

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

IN RE APPLICATION OF: Takashi NOMURA

GAU:

SERIAL NO: New Application

EXAMINER:

FILED: Herewith

FOR: METHOD AND APPARATUS FOR TRANSFERRING DATA, AND STORAGE MEDIUM

REQUEST FOR PRIORITY

ASSISTANT COMMISSIONER FOR PATENTS  
WASHINGTON, D.C. 20231



SIR:

- ☐ Full benefit of the filing date of U.S. Application Serial Number, filed, is claimed pursuant to the provisions of 35 U.S.C. §120.
- ☐ Full benefit of the filing date of U.S. Provisional Application Serial Number, filed, is claimed pursuant to the provisions of 35 U.S.C. §119(e).
- ☒ Applicants claim any right to priority from any earlier filed applications to which they may be entitled pursuant to the provisions of 35 U.S.C. §119, as noted below.

In the matter of the above-identified application for patent, notice is hereby given that the applicants claim as priority:

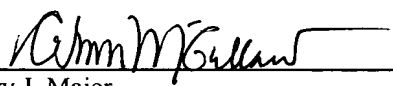
<u>COUNTRY</u>	<u>APPLICATION NUMBER</u>	<u>MONTH/DAY/YEAR</u>
Japan	2000-038871	February 16, 2000

Certified copies of the corresponding Convention Application(s)

- ☒ are submitted herewith
- ☐ will be submitted prior to payment of the Final Fee
- ☐ were filed in prior application Serial No. filed
- ☐ were submitted to the International Bureau in PCT Application Number .  
Receipt of the certified copies by the International Bureau in a timely manner under PCT Rule 17.1(a) has been acknowledged as evidenced by the attached PCT/IB/304.
- ☐ (A) Application Serial No.(s) were filed in prior application Serial No. filed ; and  
(B) Application Serial No.(s)
  - ☐ are submitted herewith
  - ☐ will be submitted prior to payment of the Final Fee

Respectfully Submitted,

OBLON, SPIVAK, McCLELLAND,  
MAIER & NEUSTADT, P.C.

  
Gregory J. Maier

Registration No. 25,599

C. Irvin McClelland  
Registration Number 21,124



22850

Tel. (703) 413-3000  
Fax. (703) 413-2220  
(OSMMN 10/98)

日 本 国 特 許 庁

PATENT OFFICE  
JAPANESE GOVERNMENT

J1046 U.S. PTO  
09/782287  
02/14/01

別紙添付の書類に記載されている事項は下記の出願書類に記載されて  
る事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed  
with this Office.

出 願 年 月 日

Date of Application:

2000年 2月16日

出 願 番 号

Application Number:

特願2000-038871

願 人

Applicant(s):

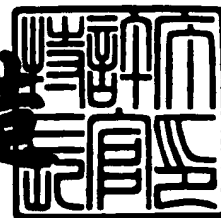
ソニー株式会社

CERTIFIED COPY OF  
PRIORITY DOCUMENT

2001年 1月 5日

特 許 庁 長 官  
Commissioner,  
Patent Office

及 川 耕 造



【書類名】 特許願

【整理番号】 0000082003

【提出日】 平成12年 2月16日

【あて先】 特許庁長官殿

【国際特許分類】 H04L 29/04

【発明者】

    【住所又は居所】 東京都品川区北品川 6 丁目 7 番 3 5 号 ソニー株式会社  
内

    【氏名】 野村 隆

【特許出願人】

    【識別番号】 000002185

    【氏名又は名称】 ソニー株式会社

    【代表者】 出井 伸之

【代理人】

    【識別番号】 100082131

    【弁理士】

    【氏名又は名称】 稲本 義雄

    【電話番号】 03-3369-6479

【手数料の表示】

    【予納台帳番号】 032089

    【納付金額】 21,000円

【提出物件の目録】

    【物件名】 明細書 1

    【物件名】 図面 1

    【物件名】 要約書 1

    【包括委任状番号】 9708842

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 データ中継装置および方法、並びに提供媒体

【特許請求の範囲】

【請求項 1】 第 1 のネットワークと第 2 のネットワークとの間でデータを中継するデータ中継装置において、

前記第 1 のネットワークを介して転送されてきた、前記第 2 のネットワークに転送するデータをフロー毎に蓄積する蓄積手段と、

前記蓄積手段による前記データのフロー毎の蓄積量を検出する検出手段と、

前記検出手段による検出結果に対応して、前記第 2 のネットワークに伝送される前記データを制御する制御手段と

を備えることを特徴とするデータ中継装置。

【請求項 2】 前記第 1 のネットワークと第 2 のネットワークは、広域デジタル網と IEEE1394 シリアルバスである

ことを特徴とする請求項 1 に記載のデータ中継装置。

【請求項 3】 前記制御手段は、前記蓄積手段における蓄積量が、予め定められている閾値以上になった後、前記蓄積手段に蓄積された前記データの読み出しを開始する

ことを特徴とする請求項 1 に記載のデータ中継装置。

【請求項 4】 前記制御手段は、前記蓄積手段における蓄積量が、予め定められている閾値以上になったとき、前記蓄積手段に蓄積された前記データのうち、非実質的なデータを廃棄する

ことを特徴とする請求項 1 に記載のデータ中継装置。

【請求項 5】 前記制御手段は、前記蓄積手段における蓄積量が、予め定められている閾値以下になったとき、前記第 2 のネットワークに転送する前記データに、非実質的なデータを挿入する

ことを特徴とする請求項 1 に記載のデータ中継装置。

【請求項 6】 前記制御手段は、前記蓄積手段に蓄積された前記データのうち、非実質的なデータを廃棄するか、または前記第 2 のネットワークに転送する前記データに、非実質的なデータを挿入するとき、前記データに含まれるタイム

スタンプのずれを、所定の期間にわたって徐々に補正する

ことを特徴とする請求項 1 に記載のデータ中継装置。

【請求項 7】 前記制御手段は、前記タイムスタンプを補正する期間の略中間の位置で、前記非実質的なデータを挿入するか、または廃棄する

ことを特徴とする請求項 6 に記載のデータ中継装置。

【請求項 8】 前記第 1 のネットワークは、ネットワーククロックが同期していない他の第 1 のネットワークと接続されている

ことを特徴とする請求項 1 に記載のデータ中継装置。

【請求項 9】 前記データは、時間的に連続した内容を有する映像データまたは音声データである

ことを特徴とする請求項 1 に記載のデータ中継装置。

【請求項 10】 第 1 のネットワークと第 2 のネットワークとの間でデータを中継するデータ中継装置のデータ中継方法において、

前記第 1 のネットワークを介して転送されてきた、前記第 2 のネットワークに転送するデータのフロー毎の蓄積を制御する蓄積制御ステップと、

前記蓄積制御ステップの処理による前記データのフロー毎の蓄積量を検出する検出ステップと、

前記検出ステップの処理による検出結果に対応して、前記データの前記第 2 のネットワークへの伝送を制御する制御ステップと

を含むことを特徴とするデータ中継方法。

【請求項 11】 第 1 のネットワークと第 2 のネットワークとの間でデータを中継するデータ中継装置を制御するプログラムにおいて、

前記第 1 のネットワークを介して転送されてきた、前記第 2 のネットワークに転送するデータのフロー毎の蓄積を制御する蓄積制御ステップと、

前記蓄積制御ステップの処理による前記データのフロー毎の蓄積量を検出する検出ステップと、

前記検出ステップの処理による検出結果に対応して、前記データの前記第 2 のネットワークへの伝送を制御する制御ステップと

を含むことを特徴とするコンピュータが読みとり可能なプログラムが記録され

ている記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、データ中継装置および方法、並びに提供媒体に関し、特にクロックが同期していないネットワーク間で連続する動画像データを欠落させることなく授受することができるようにした、データ中継装置および方法、並びに提供媒体に関する。

【0002】

【従来の技術】

図1は、従来のネットワークシステムの構成例を表している。このネットワークシステムにおいては、デジタルビデオカセットレコーダ（DVCR）11が接続されているIEEE1394高速デジタルシリアルバス（以下、単に1394シリアルバスとも称する）12が、ATM（Asynchronous Transfer Mode）／1394中継器13から、UNI（User Network Interface）14を介してATMネットワーク15に接続されている。ATMネットワーク15はさらに、UNI16を介してATM／1394中継器17に接続されている。ATM／1394中継器17には、DVCR19が接続されている1394シリアルバス18が接続されている。

【0003】

1394シリアルバス12（1394シリアルバス18も同様）においては、図2に示すようにデータが転送される。すなわち、DVCR11が伝送するデータからなるソースパケット（図2（A））は、480バイト単位 of データブロックに分割される（図2（B））。このデータブロックに、アイソクロナスパケットヘッダとCIP（Common Isochronous Packet）ヘッダが付加され、125 $\mu$ sのサイクル周期のうちの所定のタイミングのサイクルにおいて、アイソクロナスパケットとして伝送される。

【0004】

各サイクルの先頭には、1394シリアルバス12に接続されている機器のうちの所定の機器で構成されるサイクルマスタからサイクルスタートパケットが送信さ

れる。1394シリアルバス12上において同期をとるために、1394シリアルバス上に接続されている各機器は32ビットのサイクルタイムレジスタを内蔵しており、サイクルマスタの24.576MHzの周波数のリファレンスクロック（以下、バスリファレンスクロックとも称する）に同期しているサイクルスタートパケット内のサイクルタイムデータの値（サイクルマスタのサイクルタイムレジスタ値に等しい）を自身のサイクルタイムレジスタに反映させることで、125 $\mu$ s毎にサイクルタイムレジスタの値を同期させて動作する。従って、ATM/1394中継器13も、1394シリアルバス12とのインタフェース処理を行う部分において、上記のようにサイクルタイムレジスタ値を同期させながら動作する。

#### 【0005】

1394インタフェース部でインタフェース処理されたパケットデータは、ATM/1394中継器13のATMインタフェース部でATMセルに変換され、UNI14を介してATMネットワーク15に送信される。ATMネットワーク15は、そこに接続されている各機器の同期をとるために、8KHzの周波数のリファレンスクロック（以下、ATMリファレンスクロックとも称する）に同期して動作するようになっている。従って、ATM/1394中継器13のATMインタフェース部は、このATMリファレンスクロックに同期して、各種の処理を実行する。

#### 【0006】

ATMネットワーク15を介してATM/1394中継器13から伝送されてきたATMセルは、UNI16を介してATM/1394中継器17に供給される。そのATMインタフェース部は、入力されたATMセルを組み立て、1394インタフェース部に出力する。このATMインタフェース部も、ATMネットワーク15のATMリファレンスクロックに同期して動作する。ATMインタフェース部より出力されたデータは、1394インタフェース部においてパケット化され、1394シリアルバス18を介してDVCR19に供給される。ATM/1394中継器17の1394インタフェース部は、1394シリアルバス18に接続されている各機器の同期をとるためのサイクルタイムレジスタの値を同期させながら動作する。

#### 【0007】

図3は、このようにして、DVCR11が、1394シリアルバス12からATMネット

ワーク 1 5 を介して 1394 シリアルバス 1 8 の DVCR 1 9 にデータを転送する場合の原理的なタイミングチャートを表している。DVCR 1 1 が出力するデータが、例えば、NTSC 方式の画像データであるとするとき、その 2 9 . 9 7 Hz の周波数のフレーム同期信号は、2 4 . 5 7 6 MHz の周波数のバスリファレンスクロックで、例えば、時刻  $t_1$ 、 $t_4$ 、 $t_7$  においてサンプリングされる（図 3（A））。

#### 【0008】

時刻  $t_1$  で取り込まれた画像データは、DVCR 1 1 から 1394 シリアルバス 1 2 に対して、時刻  $t_2$  から始まるバスサイクルで伝送される。このとき、その CIP パケット CIP1 には、タイムスタンプが付加されている（図 3（B））。

#### 【0009】

すなわち、図 4 に示すように、1394 シリアルバスを伝送されるアイソクロナスパケットは、1394 ヘッダ、CIP ヘッダ 1、CIP ヘッダ 2、およびデータから構成されており、そのうちの CIP ヘッダ 2 には、1 6 ビットからなる時間情報（Sync Time）がタイムスタンプとして配置されている。1394 シリアルバス 1 2 に接続されている各機器は、バス上における処理の同期をとるために、サイクルタイムレジスタを内蔵しており、その下位 1 6 ビットの値として、このタイムスタンプの値を設定する。CIP パケット CIP1 のタイムスタンプは、サンプリング時（時刻  $t_1$ ）のサイクルタイムレジスタの値に、遅延加算時間  $T_{delayAddCount}$  を加算した値とされている。すなわち、タイムスタンプは、時刻  $t_1$  から遅延加算時間  $T_{delayAddCount}$  だけ経過した時刻  $t_3$  に対応した値となっている。この遅延加算時間  $T_{delayAddCount}$  は、1394 シリアルバス 1 2 のサイクルタイミングのずれなどのジッタを吸収するための時間に対応している。

#### 【0010】

受信側の 1394 シリアルバス 1 8 を介してこの CIP パケット CIP1 を受信すると、DVCR 1 9 は、そこに含まれるタイムスタンプを抽出する（図 3（C））。上述したように、このタイムスタンプの時刻は、時刻  $t_3$  に対応している。そこで、DVCR 1 9 は、時刻  $t_3$  のタイミングにおいて、第 1 のフレームのフレーム同期信号を生成する。以下同様に、第 2 のフレーム、第 3 のフレームなどにおいても、順次、同様の処理が行われる。



## 【 0 0 1 1 】

図 3 に示したタイミングチャートは、あくまで原理的なものであるが、実際のタイミングチャートは、図 5 に示すようになる。すなわち、時刻  $t_1$  で取り込まれた第 1 のフレームの同期信号は、その時のサンプリング時刻に、遅延加算時間  $T_{delayAddCount}$  を加算した時刻  $t_3$  に対応するタイムスタンプを含む CIP パケット CIP1 として、時刻  $t_2$  で 1394 シリアルバス 1 2 に伝送される。この CIP パケット CIP1 は、1394 シリアルバス 1 2、ATM/1394 中継器 1 3、UNI 1 4、ATM ネットワーク 1 5、UNI 1 6、ATM/1394 中継器 1 7、1394 シリアルバス 1 8 の各伝送路上の総合的な遅延時間  $T_{delayNet 1}$  だけ遅延され、時刻  $t_4$  から始まるバスサイクルのタイミングにおいて、DVCR 1 9 に供給される。DVCR 1 9 は、この CIP パケット CIP1 からタイムスタンプを抽出し（図 5 (C)）、そのタイムスタンプに対応する時刻  $t_6$  で、第 1 のフレームの同期信号を生成する（図 5 (D)）。

## 【 0 0 1 2 】

受信側の 1394 シリアルバス 1 8 に接続されている DVCR 1 9 が、CIP パケット CIP1 から抽出したタイムスタンプに基づいて、時刻  $t_4$  から計時した時刻  $t_6$  までの時間  $T_{offsetAddCount 1 \# 2}$  は、1394 シリアルバス 1 8 のバスリファレンスクロックに基づいて計時される。これに対して、送信側の 1394 バス 1 2 に接続されている DVCR 1 1 が、CIP パケット CIP1 のタイムスタンプに設定した時刻  $t_3$  は、1394 シリアルバス 1 2 のバスリファレンスクロックに基づいて、バスサイクルの開始時刻  $t_2$  から、時間  $T_{offsetAddCount 1 \# 1}$  だけ経過したときの時刻である（図 5 (B)）。この時間  $T_{offsetAddCount 1 \# 1}$  は、時刻  $t_3$  と時刻  $t_2$  の差（すなわち、時刻  $t_3$  から時間  $T_{delayNet 1}$  だけ経過した時刻  $t_5$  と、時刻  $t_2$  から時間  $T_{delayNet 1}$  だけ経過した時刻  $t_4$  との差）に対応している（図 5 (B)）。

## 【 0 0 1 3 】

送信側の 1394 シリアルバス 1 2 のバスリファレンスクロックと、受信側の 1394 シリアルバス 1 8 のバスリファレンスクロックとは同期していないため、1394 シリアルバス 1 2 のサイクル周期（図 5 (B)）と、1394 シリアルバス 1 8 のサイクル周期（図 5 (C)）とは正確には一致しておらず、その結果、1394 シリアルバス 1 2 におけるフレーム周期（図 5 (B)）としての時刻  $t_3$  から時刻  $t_9$  まで

の時間TsndFrameと、図5（D）に示す受信側の1394シリアルバス18におけるフレーム周期TrevFrameとは一致しない。

【0014】

その結果、DVCR11側における画像の色合いと、DVCR19側における画像の色合いとが微妙に変化したり、音色についても、送信側と受信側とで微妙にずれが生じる。

【0015】

このようなバスサイクルの時間のずれは、受信側のATM/1394中継器17のバッファのオーバーフローまたはアンダーフローを引き起こす。オーバーフローまたはアンダーフローのどちらが発生するかは、送信側と受信側のバスサイクルの相対的な関係によって決定され、送信側のバスサイクルが短い場合はオーバーフローとなり、長い場合はアンダーフローとなる。図5のタイミングチャートは、前者の場合を示しており、この場合、徐々に蓄積遅延時間が長くなるとともに、ATM/1394中継器17内に滞留するパケット数が増加する。

【0016】

ここで、オーバーフローが発生するまでの時間を算出すると、例えばATM/1394中継器17内のバッファ容量を16Mbyte、相対差を30ppm（クロックを生成するための電圧制御水晶発振回路の標準的な偏差の値）と仮定した場合、パケットのサイクルタイムレジスタの値での長さは3072であるから、

$$\begin{aligned} 1 \text{ CIPパケットが滞留する時間} T_{cip} &= 3072 / (24.576 \times 30) \\ &= 4.17 \text{ sec} \end{aligned}$$

16Mbyte（16777216byte）のバッファがオーバーフローする時間Tover

$$= T_{cip} \times 16777216 / 488 = 143248 \text{ sec} = 39.8 \text{ hour}$$

となり、およそ40時間程度で16Mbyteのバッファがオーバーフローする。

【0017】

また、アンダーフローが発生する時間は、ジッタ等を吸収するために蓄積するCIPパケット数によって増減する。蓄積パケット数を多くするとアンダーフローまでの時間は長くなるが（なかなかアンダーフローしないが）、それに比例して

、ATM/1394中継器 1 7 での遅延時間も増加してしまう。逆に、蓄積パケット数を少なくすると、遅延時間は小さくなるが、アンダーフローまでの時間も短くなってしまふ（すぐにアンダーフローしてしまう）。ここで、アンダーフローまでの時間を算出すると、例えば蓄積するCIPパケット数を 2 4 0 0 ( $125\mu\text{s} \times 2400 = 300\text{ms}$ : Real Timeアプリケーションの遅延時間の限界値)、相対差を 30 ppmと仮定した場合、

$$\begin{aligned} 1 \text{ CIPパケットが流出する時間 } T_{\text{cip}} &= 3072 / (24.576 \times 30) \\ &= 4.17 \text{ sec} \end{aligned}$$

蓄積したCIPパケットがアンダーフローする時間  $T_{\text{under}}$

$$= T_{\text{cip}} \times 2400 = 10008 \text{ sec} = 2.78 \text{ hour}$$

となり、およそ3時間程度で蓄積した2400個のCIPパケットがアンダーフローする。

#### 【0018】

そこで、本出願人は、特願平 1 1 - 1 8 0 6 5 号として、図 6 に示すようなネットワークシステムを提案した。図 6 のネットワークシステムにおいて、図 1 に対応する部分には、同一の符号を付してあり、その説明は適宜省略する。このシステムの基本的な構成は、図 1 に示した場合と同様であるが、1394シリアルバス 1 2 と ATM ネットワーク 1 5 の間に配置されている ATM/1394中継器 4 1、および ATM ネットワーク 1 5 と 1394シリアルバス 1 8 との間に配置されている ATM/1394中継器 4 2 の構成が、図 1 における場合と異なっている。

#### 【0019】

すなわち、この構成例においては、ATM/1394中継器 4 1 は、1394シリアルバス 1 2 のサイクルマスタとなっており、また、ATM/1394中継器 4 2 は、1394シリアルバス 1 8 のサイクルマスタとなっており、それぞれは、その1394バスリファレンスクロック発生器 4 1 A、4 2 Aにより、1394シリアルバス 1 2 または1394シリアルバス 1 8 で使用される 24.576 MHzの周波数のバスリファレンスクロックを、ATM ネットワーク 1 5 の ATM リファレンスクロック発生器 1 5 A が発生する 8 KHzの周波数の ATM リファレンスクロック (SDH Flame Pulse) に同期して生成するようになされている。

## 【 0 0 2 0 】

このように構成することで、1394シリアルバス 1 2 と1394シリアルバス 1 8 のクロックが同期し、アンダーフローまたはオーバーフローさせることなくパケットを転送することができる。

## 【 0 0 2 1 】

## 【発明が解決しようとする課題】

ところで、規模が大きくなると、ネットワークシステムは、2つ以上のATMネットワークを含んで構成されることがある。このようなネットワークシステムにおいては、例えば2つのATMネットワークは、NNI (Network to Network Interface) で相互に接続されているが、一方のATMネットワークのATMクロックリファレンスと、他方のATMネットワークのATMクロックリファレンスとは、独立しており、相互に同期していない。

## 【 0 0 2 2 】

従って、一方のATMネットワークとそこに接続されている1394シリアルバスのクロックを同期させ、かつ、他方のATMネットワークとそこに接続されている1394シリアルバスのクロックを同期させたとしても、一方の1394シリアルバスのクロックと他方の1394シリアルバスのクロックとは同期していない。従って例えば、一方の1394シリアルバスに接続されているDVCRから再生された動画像データを、他方の1394シリアルバスに接続されているDVCRに、2つのATMネットワークを介して転送し、記録させるような場合、オーバーフローやアンダーフローが発生し、動画像データの一部が欠落してしまう課題があった。

## 【 0 0 2 3 】

本発明はこのような状況に鑑みてなされたものであり、オーバーフローやアンダーフローを防止し、連続するデータを過不足なく授受することができるようにするものである。

## 【 0 0 2 4 】

## 【課題を解決するための手段】

本発明のデータ中継装置は、第1のネットワークを介して転送されてきた、第2のネットワークに転送するデータをフロー毎に蓄積する蓄積手段と、蓄積手段

によるデータのフロー毎の蓄積量を検出する検出手段と、検出手段による検出結果に対応して、第2のネットワークに伝送されるデータを制御する制御手段とを備えることを特徴とする。

【0025】

前記第1のネットワークと第2のネットワークは、広域デジタル網とIEEE1394シリアルバスとすることができる。

【0026】

前記制御手段には、蓄積手段における蓄積量が、予め定められている閾値以上になった後、蓄積手段に蓄積されたデータの読み出しを開始させるようにすることができる。

【0027】

前記制御手段には、蓄積手段における蓄積量が、予め定められている閾値以上になったとき、蓄積手段に蓄積されたデータのうち、非実質的なデータを廃棄させるようにすることができる。

【0028】

前記制御手段には、蓄積手段における蓄積量が、予め定められている閾値以下になったとき、第2のネットワークに転送するデータに、非実質的なデータを挿入させるようにすることができる。

【0029】

前記制御手段には、蓄積手段に蓄積されたデータのうち、非実質的なデータを廃棄するか、または第2のネットワークに転送するデータに、非実質的なデータを挿入するとき、データに含まれるタイムスタンプのずれを、所定の期間にわたって徐々に補正させるようにすることができる。

【0030】

前記制御手段には、タイムスタンプを補正する期間の略中間の位置で、非実質的なデータを挿入するか、または廃棄させるようにすることができる。

【0031】

前記第1のネットワークは、ネットワーククロックが同期していない他の第1のネットワークと接続させるようにすることができる。

## 【 0 0 3 2 】

前記データは、時間的に連続した内容を有する映像データまたは音声データとすることができる。

## 【 0 0 3 3 】

本発明のデータ中継方法は、第 1 のネットワークを介して転送されてきた、第 2 のネットワークに転送するデータのフロー毎の蓄積を制御する蓄積制御ステップと、蓄積制御ステップの処理によるデータのフロー毎の蓄積量を検出する検出ステップと、検出ステップの処理による検出結果に対応して、データの第 2 のネットワークへの伝送を制御する制御ステップとを含むことを特徴とする。

## 【 0 0 3 4 】

本発明の記録媒体に記録されているプログラムは、第 1 のネットワークを介して転送されてきた、第 2 のネットワークに転送するデータのフロー毎の蓄積を制御する蓄積制御ステップと、蓄積制御ステップの処理によるデータのフロー毎の蓄積量を検出する検出ステップと、検出ステップの処理による検出結果に対応して、データの第 2 のネットワークへの伝送を制御する制御ステップとを含むことを特徴とする。

## 【 0 0 3 5 】

本発明のデータ中継装置、データ中継方法、および記録媒体に記録されているプログラムにおいては、第 1 のネットワークを介して転送されてきた、第 2 のネットワークに転送するデータがフロー毎に蓄積され、データのフロー毎の蓄積量に対応して、データの第 2 のネットワークへの伝送が制御される。

## 【 0 0 3 6 】

## 【発明の実施の形態】

本発明のネットワークシステムは、図 7 に示すように構成される。このネットワークシステムにおいては、デジタルビデオカセットレコーダ (DVCR) 1 1 - 1 が接続されている IEEE1394 シリアルバス 1 2 - 1 が、ATM/1394 中継器 4 1 - 1 から、UNI 1 4 - 1 を介して ATM ネットワーク 1 5 - 1 に接続されている。ATM ネットワーク 1 5 - 1 はさらに、UNI 1 6 - 1 を介して ATM/1394 中継器 4 2 - 1 に接続されている。ATM/1394 中継器 4 2 - 1 には、DVCR 1 9 - 1 が接続されてい

る1394シリアルバス18-1が接続されている。

【0037】

ATM/1394中継器41-1は、1394シリアルバス12-1のサイクルマスタとなっており、また、ATM/1394中継器42-1は、1394シリアルバス18-1のサイクルマスタとなっており、それぞれは、そのバスリファレンスクロック発生器41A-1、42A-1により、1394シリアルバス12-1または1394シリアルバス18-1で使用される24.576MHzの周波数のバスリファレンスクロックを、ATMネットワーク15-1のATMリファレンスクロック発生器15A-1が発生する8KHzの周波数のATMリファレンスクロック（SDH Flame Pulse）に同期して生成するようになされている。

【0038】

同様に、デジタルビデオカセットレコーダ（DVCR）11-2が接続されているIEEE1394シリアルバス12-2が、ATM/1394中継器41-2から、UNI14-2を介してATMネットワーク15-2に接続されている。ATMネットワーク15-2はさらに、UNI16-2を介してATM/1394中継器42-2に接続されている。ATM/1394中継器42-2には、DVCR19-2が接続されている1394シリアルバス18-2が接続されている。

【0039】

ATM/1394中継器41-2は、1394シリアルバス12-2のサイクルマスタとなっており、また、ATM/1394中継器42-2は、1394シリアルバス18-2のサイクルマスタとなっており、それぞれは、そのバスリファレンスクロック発生器41A-2、42A-2により、1394シリアルバス12-2または1394シリアルバス18-2で使用される24.576MHzの周波数のバスリファレンスクロックを、ATMネットワーク15-2のATMリファレンスクロック発生器15A-2が発生する8KHzの周波数のATMリファレンスクロック（SDH Flame Pulse）に同期して生成するようになされている。

【0040】

2つのATMネットワーク15-1とATMネットワーク15-2は、NNI（Network to Network Interface）101によりで相互に接続されている。しかしなが

ら、ATMネットワーク15-1のATMクロックリファレンスとATMネットワーク15-2のATMクロックリファレンスとは、独立しており、相互に同期していない。

#### 【0041】

図8は、ATM/1394中継器42-2の構成を表している（図示は省略するが、ATM/1394中継器42-1、ATM/1394中継器41-1、41-2も同様の構成とされている）。

#### 【0042】

CPU60は、プログラムに対応してATM SAR (Segmentation and Reassembly) ブロック58、ATM PHYブロック57、IEEE1394LINKレイヤブロック55、IEEE1394 3ポートPHYブロック54、およびシステムコントローラ61を制御し、インタフェース処理を実行させる。

#### 【0043】

スケジューリングタイマ63は、8KHzの周波数のATMリファレンスクロック（ATM/SDHフレームタイミング）を分周して、CPU60に対してタイマ割り込み信号を出力する。メモリブロック62は、ローカルバス64を介してシステムコントローラ61と接続されており、送受信するパケットデータを記憶する。システムコントローラ61には、PCI (Peripheral Component Interconnect) バス65を介してATMネットワーク（ATM/SDH (Synchronous Digital Hierarchy) 網）15-2に対する送受信処理を行うブロック71と、1394シリアルバス18-2に対する送受信処理を行うブロック72とが接続されている。前者のブロック71は、ATM SARブロック58、ATM PHYブロック57、および光学 (Optics) ブロック56とにより構成され、後者のブロック72は、IEEE1394LINKレイヤブロック55、IEEE1394 3ポートPHYブロック54により構成されている。

#### 【0044】

ATM SARブロック58は、メモリブロック62からシステムコントローラ61およびPCIバス65を介して転送されてきたデータをATMセルに分割し、ATM PHYブロック57に転送する。ATM SARブロック58は、また、逆に、ATM PHYブロック57より供給されたATMセルを組み立て、システムコントローラ61およびPCI



バス 6 5 を介してメモリブロック 6 2 に出力する。ATM PHY ブロック 5 7 は、ATM SAR ブロック 5 8 より供給された ATM セルを、光学ブロック 5 6 の仕様に対応するデータに変換して光学ブロック 5 6 に出力するとともに、光学ブロック 5 6 より入力されたデータから ATM SAR ブロック 5 8 に渡すべき ATM セルを取り出して、ATM SAR ブロック 5 8 に出力する。光学ブロック 5 6 は、ATM ネットワーク 1 5 - 2 より供給された光信号によるデータを受信し、電気信号に変換して、ATM PHY ブロック 5 7 に供給するとともに、ATM PHY ブロック 5 7 より供給されたデータを光信号に変換して、ATM ネットワーク 1 5 - 2 に出力する。

#### 【 0 0 4 5 】

IEEE1394 LINK レイヤブロック 5 5 は、システムコントローラ 6 1 および PCI バス 6 5 を介してメモリブロック 6 2 より入力された ATM フォーマットのデータを、CPU 6 0 によって予め設定された 1394 シリアルバスのフォーマットに変換して、IEEE1394 3 ポート PHY ブロック 5 4 に出力するとともに、逆に、IEEE1394 3 ポート PHY ブロック 5 4 より入力された 1394 シリアルバスのフォーマットのデータを、システムコントローラ 6 1 および PCI バス 6 5 を介してメモリブロック 6 2 に出力し、CPU 6 0 によって ATM のフォーマットに変換させる。IEEE1394 3 ポート PHY ブロック 5 4 は、IEEE1394 LINK レイヤブロック 5 5 より入力されたデータを、IEEE1394 シリアルバスの信号に変換して、ポート 5 1 乃至 5 3 から 1394 シリアルバス 1 8 - 2 に出力する。また、IEEE1394 3 ポート PHY ブロック 5 4 は、ポート 5 1 乃至 5 3 を介して 1394 シリアルバス 1 8 - 2 より入力されたデータを受信し、これを IEEE1394 LINK レイヤブロック 5 5 に出力する。

#### 【 0 0 4 6 】

ATM PHY ブロック 5 7 は、8 KHz の周波数の ATM リファレンスクロックを抽出する。この ATM リファレンスクロックは、クロック同期回路 5 9 と IEEE1394 LINK レイヤブロック 5 5 に供給される。IEEE1394 LINK ブロック 5 5 は、入力された ATM リファレンスクロックに同期して、1394 シリアルバスのサイクルスタートパケットの送出タイミングを検出する。クロック同期回路 5 9 ( 図 7 のバスリファレンスクロック発生器 4 2 A - 2 に対応する ) は、ATM PHY ブロック 5 7 より入力された ATM リファレンスクロックに同期して、2 4 . 5 7 6 MHz の周波数の 1394 シリ

アルバスのためのバスリファレンスクロックを生成し、IEEE1394 3ポートPHYブロック54に出力している。

#### 【0047】

クロック同期回路59は、図9に示すように構成されている。すなわち、クロック同期回路59は、PLL回路として構成され、ATM PHYブロック57より入力された8 KHzの周波数のATMリファレンスクロックと、分周器81より入力された8 KHzの周波数のクロックとを位相比較し、その位相誤差信号をフィルタ（ローパスフィルタ）83に出力している。フィルタ83は、入力された位相誤差信号を平滑し、電圧制御水晶発振回路（VCX0）84に出力している。電圧制御水晶発振回路84は、その標準的な発振周波数が、24.576 MHzの周波数となるように設定されており、フィルタ83より入力される制御信号（制御電圧）に対応する位相のクロックを生成し、バスリファレンスクロックとしてIEEE1394 3ポートPHYブロック54に出力している。電圧制御水晶発振回路84の出力するバスリファレンスクロックはまた、分周器81に輸入され、1/3072に分周され、位相比較器82に輸入されている。

#### 【0048】

これにより、クロック同期回路59は、ATM PHYブロック57より入力される8 KHzの周波数のATMネットワーク15-2のATMリファレンスクロックに同期した、通常のIEEE1394で規定されている精度（±100 ppm以内）に比べて、より高い精度（±10 ppm以内）の、24.576 MHzの周波数の1394シリアルバスのバスリファレンスクロックを生成し、出力する。

#### 【0049】

次に、機器間のデータ送受信処理について、送信側の1394シリアルバス12-1に接続されているDVCR11-1から出力されたデジタルビデオ（DV）データを、受信側の1394シリアルバス18-2に接続されているDVCR19-2に伝送する場合を例として説明する。

#### 【0050】

図10は、送信側のDVCR11-1の処理を説明するフローチャートである。なお、この場合においても、DVCR11-1は、NTSC方式のビデオ信号をサンプリン

グし、出力するものとする。

【 0 0 5 1 】

ステップ S 1 において、DVCR 1 1 - 1 は、時刻  $t_1$  のタイミングにおいて、29.97 Hz の周波数のフレーム同期信号を自身の 24.576 MHz のクロックに基づいてサンプリングする。1394 シリアルバス 1 2 - 1 において、ATM/1394 中継器 4 1 - 1 がサイクルマスタとなっているので、1394 シリアルバス 1 2 - 1 に接続されている DVCR 1 1 - 1 のサイクルタイムレジスタ値は、ATM/1394 中継器 4 1 - 1 により生成されるサイクルスタートパケット内のサイクルタイムデータを反映させたものとなっている。サイクルタイムデータは、サイクルマスタである ATM/1394 中継器 4 1 - 1 のサイクルタイムレジスタ値そのものであり、バスリファレンスクロックによってカウントアップされる。このため、DVCR 1 1 - 1 のサイクルタイムレジスタ値は、 $125 \mu s$  毎に ATM/1394 中継器 4 1 - 1 のサイクルタイムレジスタ値と同期している。

【 0 0 5 2 】

そして、ステップ S 2 において、現在の（時刻  $t_1$  の）サイクルタイムレジスタの下位 16 ビットの値に、遅延加算時間  $T_{delayAddCount}$  を加算した値を、第 1 のフレームのフレーム同期信号を伝送する CIP パケットの CIP ヘッダの Sync Time フィールド（図 4）に格納する。そして、ステップ S 3 において、DVCR 1 1 - 1 は、その CIP パケットを、1394 シリアルバス 1 2 - 1 に、所定の時刻  $t_2$  から始まるサイクルバスのタイミングでアイソクロナスパケットとして伝送する。

【 0 0 5 3 】

従来の場合と同様に、その CIP パケットに含まれているタイムスタンプは、時刻  $t_1$  に遅延加算時間  $T_{delayAddCount}$  を加算した時刻  $t_3$  に対応した値となっている。

【 0 0 5 4 】

DVCR 1 1 - 1 が出力したデータは、1394 シリアルバス 1 2 - 1 を介して、ATM/1394 中継器 4 1 - 1 に伝送される。ATM/1394 中継器 4 1 - 1 において、このデータは、例えば、ポート 5 1（図 8 は、ATM/1394 中継器 4 2 - 2 としてだけでなく、ATM/1394 中継器 4 1 - 1 としてもその説明に用いられる）から、IEE

E1394 3ポートPHYブロック54に入力され、所定のインタフェース処理が行われる。この処理は、クロック同期回路59が出力するATMリファレンスクロックに同期したバスリファレンスクロックに基づいて行われるので、ATMネットワーク15-1における処理と位相的に同期した処理となる。IEEE1394LINKレイヤブロック55は、IEEE1394 3ポートPHYブロック54より入力されたデータを、ATM PHYブロック57より入力されたATMリファレンスクロックに同期したタイミングで生成したサイクルスタートパケット送出タイミングを基に処理し、PCIバス65に出力する。システムコントローラ61は、PCIバス65を介して入力されたデータを、ローカルバス64を介してメモリブロック62に供給し、記憶させる。

#### 【0055】

CPU60は、システムコントローラ61を介して、メモリブロック62に記憶されたデータを読み出し、ATM SARブロック58に供給する。ATM SARブロック58は、メモリブロック62から伝送されてきたデータをATMセルに分割し、ATM PHYブロック57に供給する。ATM PHYブロック57は、ATM SARブロック58より供給されたATMセルを光学ブロック56の仕様に対応するデータに変換し、光学ブロック56に出力する。光学ブロック56は、入力されたデータをUNI14-1を介してATMネットワーク15-1に出力する。

#### 【0056】

ATMネットワーク15-1は、UNI14-1を介して、ATM/1394中継器41-1より供給されたATMセルをNNI101を介してATMネットワーク15-2に供給する。ATMネットワーク15-2は、NNI101を介して供給されたATMセルをUNI16-2を介して、ATM/1394中継器42-2に供給する。

#### 【0057】

次に、図11のフローチャートを参照して、ATM/1394中継器42-2の動作について説明する。ステップS51において、CPU60は、スケジューリングタイマ63の4msの周期のスケジューリングタイマ割り込み処理により、このフローチャートに示す処理を開始する。

#### 【0058】

ここで、この割り込み処理の間隔について、図 1 2 を参照して説明する。詳細は後述するが、この実施の形態の場合、タイムスタンプを書き換えるときは、ユーザが予めデフォルトとして設定したDef\_SyncPeriod（例えば、24 フレーム）の期間にわたって、タイムスタンプが徐々に書き換えられる。1 フレーム分のDVデータは、IEEE1394の250サイクルで伝送される。1サイクルは125 $\mu$ sである。そして、図 1 1 のフローチャートの処理は、4 msの周期、すなわち32サイクルの周期で行われる。

#### 【 0 0 5 9 】

ステップ S 5 2 において、CPU 6 0 は、ATMネットワーク 1 5 - 2 から供給されたATMセルを、CIPパケットに変換し、各フロー用中間バッファに振り分け、記憶させる処理を実行する。

#### 【 0 0 6 0 】

すなわち、光学ブロック 5 6 は、ATMネットワーク 1 5 - 2 により供給されたATMセルの光信号を電気信号に変換して、ATM PHYブロック 5 7 に供給する。ATM PHYブロック 5 7 は、光学ブロック 5 6 より入力されたデータから、ATM SARブロック 5 8 に渡すべきATMセルを抽出し、ATM SARブロック 5 8 に供給する。ATM SARブロック 5 8 は、ATM PHYブロック 5 7 より供給されたATMセルを組み立て、システムコントローラ 5 1 及びPCIバス 6 5 を介してメモリブロック 6 2 に供給し、記憶させる。

#### 【 0 0 6 1 】

メモリブロック 6 2 には、例えば、図 1 3 に示すような、各フロー用の中間バッファが形成される。図 1 3 の例においては、UNI 1 6 - 2 を構成するATMファイバ 1 1 1 を介して、ATMネットワーク 1 5 - 2 から供給されるVCC (Virtual Channel Connection) 毎に、フロー用中間バッファが形成される。図 1 3 の例の場合、VCC # 1, VCC # 2 の2つのVCCが存在するため、それぞれに対応して、フロー # 1 用中間バッファ 1 2 1 - 1 とフロー # 2 用中間バッファ 1 2 1 - 2 が形成されている。例えば、VCC # 1 は、1394シリアルバス 1 8 - 1 と、1394シリアルバス 1 2 - 2 との間で形成されるVCCに対応しており、VCC # 2 は、1394シリアルバス 1 2 - 1 と、1394シリアルバス 1 8 - 2 との間に形成されるVCCに対応して

いる。従って、この例の場合、1394シリアルバス18-1から供給されたデータは、フロー#1用中間バッファ121-1に供給、記憶され、1394シリアルバス12-1より供給されたデータは、フロー#2用中間バッファ121-2に供給、記憶される。なお、ここでは説明を簡単にするため、各1394シリアルバス上のデータストリームを1本としているが、当然複数本のデータストリームを転送することも可能である。この場合、対応するVCC並びにフロー#n用中間バッファも、フローに応じた数だけ生成される。

## 【0062】

ステップS53において、CPU60は、処理フローに対応する変数nに値1を初期設定する。そしてステップS54において、CPU60は、データフローの開始時（データフローの処理を開始した直後）であり、且つフロー#n（図13の例の場合、n=1または2）用中間バッファのバッファ量（データ記憶量）がジッタ吸収閾値未満であるか否かを判定する。

## 【0063】

すなわち、図13に示されるように、各フロー用中間バッファ121-1, 121-2は、その最大バッファサイズがn（例えば、CIPを256個記憶できる容量）とされ、それぞれに、上限閾値H（例えば、CIPを96個記憶できる容量）、下限閾値L（例えば、CIPを32個記憶できる容量）、ジッタ吸収閾値J（例えば、CIPを64個記憶できる容量）が設定されている。上限閾値Hの値は、下限閾値Lより大きく、ジッタ吸収閾値Jは、上限閾値Hと下限閾値Lの間の値とされる。

## 【0064】

データフローの開始時であり、且つ、フロー#n（今の場合、n=1）用中間バッファ121-1のバッファ量がジッタ吸収閾値J未満である場合、ステップS63にスキップし、ステップS63において、全ての中間バッファ（図13の例の場合、2個の中間バッファ121-1, 121-2）からのデータの取り出しが完了したか否かが判定される。今の場合、まだ、取り出しが完了していないので、ステップS64に進み、変数nの値が1だけインクリメントされ、n=2

とされる。

【 0 0 6 5 】

そしてステップ S 5 4 に戻り、再び、今、データフローの開始時であり、且つフロー # 2 用中間バッファ 1 2 1 - 2 のバッファ量がジッタ吸収閾値 J 未満であるか否かが判定される。データフローの開始時であり、且つフロー # 2 用中間バッファのバッファ量がジッタ吸収閾値 J 未満である場合には、ステップ S 6 3 にスキップし、全ての中間バッファからの取り出し処理が完了したか否かが判定される。今の場合、まだ完了していないので、再びステップ S 6 4 に戻り、変数 n の値が 1 だけインクリメントされる。ただしこの変数 n の値は、最大値まで達した場合、次のインクリメント時には、最初の値（この例の場合 1）に戻される。この例の場合、n が 2 で最大であるので、n の値は 1 に戻される。その後、ステップ S 5 4 に戻り、それ以降の処理が繰り返し実行される。

【 0 0 6 6 】

以上のようにして、フロー # 1 用中間バッファ 1 2 1 - 1 と、フロー # 2 用中間バッファ 1 2 1 - 2 のそれぞれに、ジッタ吸収閾値 J 以上のデータが蓄積されるまで、1394 シリアルバス上へのデータの転送が待機される。これにより、ネットワーク上のジッタを効果的に吸収することが可能となる。ただし、この閾値 J を大きくすると、それだけ時間的遅延が長くなる。逆にこの閾値 J を小さくし過ぎると、吸収可能なジッタが小さくなる。

【 0 0 6 7 】

以上のような処理が繰り返されている間に、フロー # 1 用中間バッファ 1 2 1 - 1 とフロー # 2 用中間バッファ 1 2 1 - 2 には、ジッタ吸収閾値 J 以上のデータが蓄積される。この時、ステップ S 5 4 において、バッファ量がジッタ吸収閾値 J 未満ではないと判定され、ステップ S 5 5 に進む。ステップ S 5 5 において、CPU 6 0 は、フロー # n（いまの場合、n = 1）用中間バッファ 1 2 1 - 1 からデータパケット（CIP）を取り出し、ステップ S 5 6 において、その CIP がフレームの先頭の CIP であるのか否かを判定する。フレームの先頭の CIP である場合には、ステップ S 5 7 に進み、タイムスタンプ書き換え処理を実行する。このタイムスタンプ書き換え処理の詳細は、図 1 4 を参照して後述するが、CIP ヘッダの

タイムスタンプは、フレームの先頭のCIPにのみ有効な値が格納されている。そこでステップS 5 6で、フレームの先頭のCIPであると判定された場合には、ステップS 5 7において、そのCIPのタイムスタンプを書き換える処理が実行される。フレームの先頭以外のCIPの場合には、そのタイムスタンプは、無効な値のタイムスタンプであり、それを書き換える必要がないので、ステップS 5 7の処理はスキップされる。

## 【 0 0 6 8 】

次にステップS 5 8に進み、CPU 6 0は、4 ms (= 3 2 サイクル) 分のCIPの取り出しが完了したか否かを判定する。すなわち、図 1 2 を参照して説明したように、この図 1 1 のフローチャートの処理は、4 ms 周期で行われる。従って、フロー # 1 用中間バッファ 1 2 1 - 1 には、4 ms (= 3 2 サイクル) 分のデータが蓄積されていることになる。ステップS 5 8において、まだ4 ms 分のCIPの取り出しが完了していないと判定された場合には、ステップS 5 5 に戻り、それ以降の処理が繰り返し実行される。

## 【 0 0 6 9 】

ステップS 5 8において、4 ms 分のCIPの取り出しが完了したと判定された場合、ステップS 5 9に進み、CPU 6 0は、フロー # 1 用中間バッファ 1 2 1 - 1 のバッファ量を算出する。そしてステップS 6 0において、ステップS 5 9で算出したバッファ量が上限閾値H以上であるか、下限閾値L以下であるか、または上限閾値Hより小さく、かつ、下限閾値Lより大きいと判定する。ステップS 6 0において、バッファ量が上限閾値H以上であると判定された場合、そのフローの中間バッファがオーバーフローする恐れがあるので、ステップS 6 1に進み、CPU 6 0は、オーバーフロー防止処理を実行する。このオーバーフロー防止処理の詳細は、図 1 6 のフローチャートを参照して後述する。ステップS 6 0において、バッファ量が下限閾値L以下であると判定された場合には、アンダーフローの恐れがあるので、ステップS 6 2に進み、CPU 6 0はアンダーフロー防止処理を実行する。このアンダーフロー防止処理の詳細は、図 1 7 を参照して後述する。

## 【 0 0 7 0 】



ステップ S 6 1 のオーバーフロー防止処理、またはステップ S 6 2 のアンダーフロー防止処理が完了するか、またはステップ S 6 0 において、バッファ量が下限閾値 L より大きく、上限閾値 H より小さいと判定された場合、ステップ S 6 3 に進み、全ての中間バッファからの取り出し処理が完了したか否かが判定される。また、いまの場合、 $n = 1$  であり、取り出していない中間バッファが存在するので、上述したように、ステップ S 6 4 に進み、変数  $n$  を 1 だけインクリメントして、 $n = 2$  とし、ステップ S 5 4 に戻り、そのフローに関し、同様の処理が実行される。

## 【 0 0 7 1 】

ステップ S 6 3 において、全ての中間バッファからのデータの取り出しが完了したと判定された場合、ステップ S 6 5 に進み、CPU 6 0 は、取り出した CIP の 1394 シリアルバスへの送信処理を実行する。

## 【 0 0 7 2 】

すなわち、CPU 6 0 は、ステップ S 5 5 の処理で取り出した CIP をシステムコントローラ 6 1 を介して PCI バス 6 5 から IEEE1394 LINK レイヤブロック 5 5 に供給する。IEEE1394 LINK レイヤブロック 5 5 は、入力された ATM のフォーマットのデータを CPU 6 0 によって予め設定されている 1394 シリアルバスのフォーマットに変換し、IEEE1394 3 ポート PHY ブロック 5 4 に出力する。IEEE1394 3 ポート PHY ブロック 5 4 は、IEEE1394 LINK レイヤブロック 5 5 より入力されたデータを IEEE1394 シリアルバスの信号に変換し、例えば、ポート 5 1, 5 2 から 1394 シリアルバス 1 2 - 2, 1 8 - 2 に出力する。1394 シリアルバス 1 8 - 2 のアイソクロナスパケットは、DVCR 1 9 - 2 に供給される。

## 【 0 0 7 3 】

次に、図 1 4 のフローチャートを参照して、図 1 1 のステップ S 5 7 のタイムスタンプの書き換え処理について説明する。

## 【 0 0 7 4 】

最初にステップ S 8 1 において、CPU 6 0 は、その CIP に含まれる SyncTime (図 4) に格納されたタイムスタンプを取り出す。そしてステップ S 8 2 において、CPU 6 0 は、DiffCycleCount を既に算出済みであるか否かを判定する。この DiffC

ycleCountは、送信側の機器が接続されている1394シリアスバス（今の例の場合、1394シリアルバス12-1）と、受信側の機器が接続されている1394シリアルバス（今の例の場合、1394シリアスバス18-2）の、それぞれのサイクルマスタにおけるサイクルタイムレジスタの値の差分を反映した変数であり、タイムスタンプと同様に、16ビットで表され、上位4ビットは、サイクルカウント（Cycle Count）値を表し、下位12ビットは、サイクルオフセット（Cycle Offset）値を示す。

#### 【0075】

DiffCycleCountがまだ算出されていない場合には、ステップS83に進み、CPU60は、取り出されたタイムスタンプのサイクルカウント値（上位4ビット）と、そのCIPが1394シリアルバスに実際に送出されたときのサイクルタイムレジスタのCycle Count値との差分を算出する。そしてステップS84において、CPU60は、ステップS83で算出した差分に、TdelayAddのサイクルカウント値を加算することで、DiffCycleCountを演算する。その後、処理はステップS95に進む。

#### 【0076】

ステップS82において、DiffCycleCountの値が既に算出済みであると判定された場合、ステップS85に進み、CPU60は、アンダーフローが発生していることを表すフラグF\_UnderFlowOccurがオンとされているか否かを判定する。なお、このフラグは、後述する図17のステップS140で、バッファ量が下限閾値Lより小さいと判定された時にオンされる。このフラグF\_UnderFlowOccurがオンであると、ステップS85において判定された場合、ステップS86に進み、CPU60は、Sam\_SyncPeriodに、 $(3072 / \text{Def\_SyncPeriod})$ を加算する処理を実行する。

#### 【0077】

Def\_SyncPeriodは、複数フレームに分けてタイムスタンプの書き換えを徐々に行う際のフレーム数（例えば、24フレーム）を定義する定数であり、ユーザにより、デフォルトとして予め定められる。また、Sam\_SyncPeriodは、Def\_SyncPeriod中の各フレームで、ずらすべきタイムスタンプの下位12ビットの値を表す

変数である。12ビットにより最大4096まで表すことが可能であるが、その値の3072により1サイクル分（タイムスタンプの上位4ビットの最下位ビットの1）を表すものとされる。従って、図15に示すように、1サイクル分の値3072をDef\_SyncPeriod（フレーム数）で割算した値は、タイムスタンプの値をDef\_SyncPeriodで表されるフレーム数に渡って徐々に書き換えようとした場合における1フレーム分の変更値を意味する。換言すれば、Sam\_SyncPeriodの値は、Def\_SyncPeriodの期間、各フレーム毎に、 $(3072 / \text{Def\_SyncPeriod})$ ずつインクリメントされるので、Sam\_SyncPeriodの値は、Sam\_SyncPeriod1, Sam\_SyncPeriod2, ...と、各フレーム毎に、次第に大きくなる。従って、Sam\_SyncPeriodは、Def\_SyncPeriodの期間の位置（位相）を表すことになる。

#### 【0078】

ステップS87において、CPU60は、Sam\_SyncPeriodの値が $3072 / 2$ の値と等しくなったか否かを判定する。このことは、現在の処理タイミングの位置（位相）が、Def\_SyncPeriodで規定される期間（24フレーム）の中間の位置（12フレームの位置）に達したか否かを判定していることになる。Sam\_SyncPeriodの値が $3072 / 2$ の値と等しいと判定された場合、ステップS88に進み、CPU60は、F\_EmptyInsertフラグをオンさせる。このフラグF\_EmptyInsertがオンされると、図17を参照して後述するように、ステップS133において、エンプティパケットが1個、そのフロー#n用中間バッファに挿入される。ステップS87において、Sam\_SyncPeriodの値が、 $3072 / 2$ と等しくないと判定された場合、ステップS88の処理はスキップされる。すなわちこの場合には、フラグF\_EmptyInsertは、オンされない。その結果、後述する図17のステップS133のエンプティパケット挿入処理は実行されないことになる。

#### 【0079】

すなわち、この例においては、アンダーフローが発生する恐れがある場合には、図15に示すように、Def\_SyncPeriodによりユーザが予め設定したフレーム数の調度中間の位置（ $3072 / 2$ の位置）で、エンプティパケットを1個挿入する処理が実行されるようになされている。これにより、Def\_SyncPeriodで規定される期間内に、2以上のエンプティパケットが挿入されることが防止される。

## 【 0 0 8 0 】

次に、ステップ S 8 9 に進み、CPU 6 0 は、DiffCycleCount に Sam\_SyncPeriod を加算する。すなわち、ステップ S 8 6 の処理で、Sam\_SyncPeriod の値が 1 フレーム分だけインクリメントされているので、その値を DiffCycleCount に加算し、更新した DiffCycleCount の値を、ステップ S 9 5 において、CIP の元のタイムスタンプに加算することで、フレームの先頭のタイムスタンプを書き換えるのである。これにより、エンプティパケットの挿入は Def\_SyncPeriod の中間位置において 1 回だけ行われるのであるが、タイムスタンプは、Def\_SyncPeriod の全期間にわたって、各フレーム毎に徐々に書き換えられる。

## 【 0 0 8 1 】

このように、Def\_SyncPeriod を定め、その期間内に 1 回しかエンプティパケットの挿入処理を行わないようにすることで、2 回以上行うのを許容する場合に較べ、エンプティパケットの挿入に伴って行われるタイムスタンプの書き換えの変化分の値を小さくすることができる。また、そのタイムスタンプの書き換えを、Def\_SyncPeriod の全期間にわたって、各フレーム毎に、行うようにすることで、1 個のエンプティパケットの挿入に伴って発生するタイムスタンプの変化分を、Def\_SyncPeriod の期間の全体に分散することができ、1 回あたりの変化分を、さらに小さくすることができる。さらに、エンプティパケットの挿入処理を、Def\_SyncPeriod の期間（タイムスタンプを書き換える期間）のほぼ中央の位置（ $3072/2$  の位置）で行うようにすることで、1394 シリアルバスに接続されている機器に、より安定してリファレンスクロックを再生させることが可能となる。

## 【 0 0 8 2 】

ステップ S 8 5 において、フラグ F\_UnderFlowOccur がオンではないと判定された場合、ステップ S 9 0 に進み、フラグ F\_OverflowOccur がオンであるか否かが判定される。このフラグ F\_OverflowOccur は、後述する図 1 6 のステップ S 1 2 0 において、バッファ量が上限閾値 H 以上であると判定された場合に、オンとされる。

## 【 0 0 8 3 】

フラグ F\_OverflowOccur がオンであると判定された場合、ステップ S 9 1 に進

み、CPU 6 0 は、Sam\_SyncPeriodに  $(3072 / \text{Def\_SyncPeriod})$  を加算する。  
そして、ステップ S 9 2 において、ステップ S 9 1 で更新したSam\_SyncPeriodの値が  $3072 / 2$  の値と等しいか否かを判定する。このステップ S 9 1 とステップ S 9 2 の処理は、上述したステップ S 8 6 とステップ S 8 7 の処理と同様の処理である。

#### 【0084】

そしてステップ S 9 2 において、Sam\_SyncPeriodの値が  $3072 / 2$  の値と等しいと判定された場合、ステップ S 9 3 に進み、CPU 6 0 は、F\_EmptyDiscardフラグをオンさせる。このF\_EmptyDiscardフラグがオンされると、後述する図 1 6 のステップ S 1 1 3 において、エンプティパケットが 1 個だけ廃棄される処理が実行される。ステップ S 9 2 において、Sam\_SyncPeriodの値が  $3072 / 2$  と等しくないと判定された場合、ステップ S 9 3 の処理はスキップされる。

#### 【0085】

その後、ステップ S 9 4 に進み、CPU 6 0 は、ステップ S 9 1 で更新したSam\_SyncPeriodの値をDiffCycleCountから減算する。すなわちステップ S 9 1 でSam\_SyncPeriodの値を更新しているので、DiffCycleCountの値もその分だけ変更するのである。そして、ステップ S 9 5 で、DiffCycleCountの値を、元のタイムスタンプに加算し、加算した値をCIPのSync Timeにタイムスタンプとして格納する処理が行われる。

#### 【0086】

すなわち、オーバーフローが発生するおそれがある場合、エンプティパケットを廃棄するのであるが、その廃棄処理は、挿入処理と同様に、Def\_SyncPreiodで規定される期間の中間の位置で 1 回だけ行われる。ただし、それに伴うタイムスタンプの書き換えは、Def\_SyncPreiodの全期間にわたって、各フレーム毎に、徐々に行われる。

#### 【0087】

ステップ S 8 4 の処理の後、またはステップ S 9 0 において、フラグF\_OverflowOccurがオンされていないと判定された場合にも、処理はステップ S 9 5 に進む。

## 【 0 0 8 8 】

この場合、CPU 6 0 は、ステップ S 8 4 で求められたDiffCycleCountの値、またはすでに算出済みのDiffCycleCountの値を、ステップ S 8 1 の処理で取り出されたタイムスタンプの値に加算し、図 1 1 のステップ S 5 5 で取り出されたCIPのSync Timeに格納する。その後、処理は、図 1 1 のステップ S 5 8 に進む。

## 【 0 0 8 9 】

次に、図 1 6 のフローチャートを参照して、図 1 1 のステップ S 6 1 におけるオーバーフロー防止処理の詳細について説明する。この処理は、図 1 1 の処理を参照して説明したように、ステップ S 6 0 において、バッファ量が上限閾値H以上であると判定された場合に開始される。

## 【 0 0 9 0 】

ステップ S 1 1 1 において、CPU 6 0 は、F\_OverFlowOccurフラグがオンとなっているか否かを判定し、オンとなっていない場合には、ステップ S 1 1 9 に進み、T\_OverFlowPeriodがDef\_OverFlowPeriodと等しいか、それより大きいかなどかを判定される。このT\_OverFlowPeriodは、F\_OverFlowOccurフラグがオンされてからの期間を、4 ms ( 3 2 サイクル ) を単位として表すものであり、Def\_OverFlowPeriodは、ユーザがデフォルトとして予め設定する 4 ms ( 3 2 サイクル ) を単位とする値である。上述したDef\_SyncPeriodと、Sam\_SyncPeriodが、NTSC方式、あるいはPAL方式のビデオ信号のフレームを単位とするものであるのに対して、T\_OverFlowPeriodとDef\_OverFlowPeriodをサイクルを単位として規定することで、エンパティパケットの廃棄タイミングをより細かく制御し、且つ、Def\_SyncPeriodとSam\_SyncPeriodだけで指定する場合に比べて、エンパティパケットの廃棄間隔をより広くすることができる。

## 【 0 0 9 1 】

ステップ S 1 1 9 において、T\_OverFlowPeriodがDef\_OverFlowPeriodより小さいと判定された場合、ステップ S 1 2 1 に進み、CPU 6 0 は、T\_OverFlowPeriodに 3 2 サイクルを加算する。これにより、T\_OverFlowPeriodに、バッファ容量が上限閾値H以上になったときからの期間が設定されることになる。

## 【 0 0 9 2 】

以上のような処理が 3 2 サイクルを単位として繰り返されると、T\_OverFlowPeriod の値が次第に大きくなり、遂には、Def\_OverFlowPeriod と等しいか、それより大きい値になる。この時、ステップ S 1 1 9 からステップ S 1 2 0 に進み、CPU 6 0 は、F\_OverFlowOccur フラグをオンする。すなわち、このフラグは、バッファ容量が上限閾値 H 以上になってからの期間 T\_OverFlowPeriod が、ユーザが設定した期間 Def\_OverFlowPeriod より大きくなったことを表している。

## 【 0 0 9 3 】

以上のようにして、F\_OverFlowOccur フラグがオンされると、次の 3 2 サイクル後の処理のタイミングにおいて、ステップ S 1 1 1 で、そのフラグがオンされていると判定されるので、ステップ S 1 1 2 に進み、F\_EmptyDiscard フラグがオンされているか否かが判定される。このフラグは、上述したように、図 1 4 のステップ S 9 3 でオンされるものである。Sam\_SyncPeriod の値が、 $3072/2$  の値と等しくなっていないければ、この F\_EmptyDiscard フラグは、まだオンされていないことになる。このような場合には、オーバーフロー防止処理では、その後、特に処理は行われぬ。

## 【 0 0 9 4 】

ステップ S 1 1 2 において、F\_EmptyDiscard フラグにオンされていると判定された場合、ステップ S 1 1 3 に進み、CPU 6 0 は、フロー # n 用中間バッファからエンプティパケットを 1 個廃棄する処理を実行する。そして、CPU 6 0 は、エンプティパケットを廃棄したので、ステップ S 1 1 4 において、F\_EmptyDiscard フラグをオフする。

## 【 0 0 9 5 】

次に、ステップ S 1 1 5 に進み、CPU 6 0 は、Sam\_SyncPeriod の値が  $3072$  の値と等しいか、それより大きくなったか否かを判定する。Sam\_SyncPeriod の値が  $3072$  より小さい場合には、オーバーフロー防止処理では、その後、特に処理は行われぬ。

## 【 0 0 9 6 】

F\_OverFlowOccur がオンになっている場合、上述した図 1 4 のステップ S 9 0 , S 9 1 の処理により、Sam\_SyncPeriod の値は、各フレーム毎に順次大きくなる

。そして、Def\_SyncPeriodで規定するフレーム数に達したとき、Sam\_SyncPeriodの値は、3 0 7 2と等しくなる。処理対象フレームがさらに先のフレームに移行すれば、その値は3 0 7 2よりさらに大きくなる。このとき、ステップS 1 1 5から、ステップS 1 1 6に進み、CPU 6 0は、F\_OverflowOccurフラグをオフし、ステップS 1 1 7とステップS 1 1 8において、それぞれSam\_SyncPeriodの値と、T\_OverflowPeriodの値を、それぞれ0にリセットする。

#### 【 0 0 9 7 】

その後、処理は、図 1 1 のステップS 6 3に進む。

#### 【 0 0 9 8 】

次に、図 1 7 のフローチャートを参照して、図 1 1 のステップS 6 2におけるアンダーフロー防止処理の詳細について説明する。この処理は、バッファ量が下限閾値L以下になったと判定された場合に開始されるものである。

#### 【 0 0 9 9 】

ステップS 1 3 1において、CPU 6 0は、F\_UnderFlowOccurフラグがオンされているか否かを判定する。このフラグがオンされていないと判定された場合、ステップS 1 3 9に進み、T\_UnderFlowPeriodがDef\_UnderFlowPeriodと等しいか、それより大きいかが判定される。このT\_UnderFlowPeriodは、F\_UnderFlowOccurフラグがオンされてからの期間を、4 ms (3 2 サイクル) を単位として表すものであり、Def\_UnderFlowPeriodは、ユーザがデフォルトとして予め設定する4 ms (3 2 サイクル) を単位とする値である。上述したDef\_SyncPeriodと、Sam\_SyncPeriodが、NTSC方式、あるいはPAL方式のビデオ信号のフレームを単位とするものであるのに対して、T\_UnderFlowPeriodとDef\_UnderFlowPeriodをサイクルを単位として規定することで、エンプティパケットの挿入タイミングをより細かく制御し、且つ、Def\_SyncPeriodとSam\_SyncPeriodだけで指定する場合に比べて、エンプティパケットの挿入間隔をより広げることができる。

#### 【 0 1 0 0 】

ステップS 1 3 9において、T\_UnderFlowPeriodがDef\_UnderFlowPeriodより小さいと判定された場合、ステップS 1 4 1に進み、CPU 6 0は、T\_UnderFlowPeriodに3 2 サイクルを加算する。これにより、T\_UnderFlowPeriodに、バッファ容



量が下限閾値L以下になったときからの期間が設定されることになる。

【0101】

以上のような処理が32サイクルを単位として繰り返されると、T\_UnderFlowPeriodの値が次第に大きくなり、遂には、Def\_UnderFlowPeriodと等しいか、それより大きい値になる。この時、ステップS139からステップS140に進み、CPU60は、F\_UnderFlowOccurフラグをオンする。すなわち、このフラグは、バッファ容量が下限閾値L以下になってからの期間T\_UnderFlowPeriodが、ユーザが設定した期間Def\_UnderFlowPeriodより大きくなったことを表している。

【0102】

以上のようにして、F\_UnderFlowOccurフラグがオンされると、次の32サイクル後の処理のタイミングにおいて、ステップS131で、そのフラグがオンされていると判定されるので、ステップS132に進み、F\_EmptyInsertフラグがオンされているか否かが判定される。このフラグは、上述したように、図14のステップS88でオンされるものである。Sam\_SyncPeriodの値が、 $3072/2$ の値と等しくなっていないければ、このF\_EmptyInsertフラグは、まだオンされていないことになる。このような場合には、アンダーフロー防止処理では、その後、特に処理は行われない。

【0103】

ステップS132において、F\_EmptyInsertフラグにオンされていると判定された場合、ステップS133に進み、CPU60は、フロー#n用中間バッファにエンptyパケットを1個挿入する処理を実行する。そして、CPU60は、エンptyパケットを挿入したので、ステップS134において、F\_EmptyInsertフラグをオフする。

【0104】

次に、ステップS135に進み、CPU60は、Sam\_SyncPeriodの値が $3072$ の値と等しいか、それより大きくなったか否かを判定する。Sam\_SyncPeriodの値が $3072$ より小さい場合には、アンダーフロー防止処理では、その後、特に処理は行われない。

【0105】

F\_OverFlowOccurがオンになっている場合、上述した図14のステップS87、S88の処理により、Sam\_SyncPeriodの値は、各フレーム毎に順次大きくなる。そして、Def\_SyncPeriodで規定するフレーム数に達したとき、Sam\_SyncPeriodの値は、3072と等しくなる。処理対象フレームがさらに先のフレームに移行すれば、その値は3072よりさらに大きくなる。このとき、ステップS135から、ステップS136に進み、CPU60は、F\_UnderFlowOccurフラグをオフし、ステップS137とステップS138において、それぞれSam\_SyncPeriodの値と、T\_UnderFlowPeriodの値を、それぞれ0にリセットする。

【0106】

その後、処理は、図11のステップS63に進む。

【0107】

DVCR19-2は、1394シリアルバス18-2を介してアイソクロナスパケットとして、CIPパケットが供給されると、図18のフローチャートに示す処理を実行する。

【0108】

最初にステップS151において、DVCR19-2は、フレームの先頭のCIPを検出する処理を実行する。さらに、DVCR19-2は、ステップS152において、CIPのヘッダのSync Timeフィールドに格納されているタイムスタンプを抽出する。なお、このSync Timeフィールドは、フレームの先頭のCIPにおいてのみ有効である。

【0109】

次に、ステップS153において、ステップS152で抽出したタイムスタンプの値と、そのCIPを受信したときのサイクルタイムレジスタの値（下位16ビット）との差分が演算される。そこで、ステップS154において、DVCR19-2は、ステップS153で求めた差分値に基づくタイミングにおいて、第1のフレームのフレーム同期信号を再生する。

【0110】

以上においては、バスとして1394シリアルバスを用い、ネットワークとしてATMネットワークを用いるようにしたが、その他のバスまたはネットワークを用い

ることも可能である。

【 0 1 1 1 】

なお、本明細書において、システムとは、複数の装置により構成される装置全体を表すものである。

【 0 1 1 2 】

上述した一連の処理は、ハードウェアにより実行させることもできるが、ソフトウェアにより実行させることもできる。一連の処理をソフトウェアにより実行させる場合には、そのソフトウェアを構成するプログラムが、専用のハードウェアに組み込まれているコンピュータ、または、各種のプログラムをインストールすることで、各種の機能を実行することが可能な、例えば汎用のパーソナルコンピュータなどに、記録媒体からインストールされる。

【 0 1 1 3 】

この記録媒体は、コンピュータとは別に、ユーザにプログラムを提供するために配布される、プログラムが記録されている磁気ディスク（フロッピディスクを含む）、光ディスク（CD-ROM(Compact Disk-Read Only Memory),DVD(Digital Versatile Disk)を含む）、光磁気ディスクMD（Mini-Disk）を含む）、もしくは半導体メモリなどよりなるパッケージメディアにより構成されるだけでなく、コンピュータに予め組み込まれた状態でユーザに提供される、プログラムが記録されているROMや、ハードディスクなどで構成される。

【 0 1 1 4 】

なお、本明細書において、記録媒体に記録されるプログラムを記述するステップは、記載された順序に沿って時系列的に行われる処理はもちろん、必ずしも時系列的に処理されなくとも、並列的あるいは個別に実行される処理をも含むものである。

【 0 1 1 5 】

【発明の効果】

以上の如く、本発明のデータ中継装置、データ中継方法、および記録媒体に記録されているプログラムによれば、第1のネットワークを介して転送されてきた、第2のネットワークに転送するデータをフロー毎に蓄積し、データのフロー毎

の蓄積量に対応して、データの第2のネットワークへの伝送を制御するようにしたので、オーバーフローやアンダーフローを防止し、連続するデータをリアルタイムで過不足なく授受することが可能となる。

【図面の簡単な説明】

【図1】

従来のネットワークシステムの構成を示す図である。

【図2】

1394シリアルバスのアイソクロナスパケットの転送を説明する図である。

【図3】

図1のネットワークシステムの動作原理を説明するタイミングチャートである。

【図4】

アイソクロナスパケットの構成を示す図である。

【図5】

図1のネットワークシステムの動作を説明するタイミングチャートである。

【図6】

先に提出したネットワークシステムの構成を示す図である。

【図7】

本発明を適用したネットワークシステムの構成例を示す図である。

【図8】

図7のATM/1394中継器42-2の構成を示すブロック図である。

【図9】

図8のクロック同期回路59の構成を示すブロック図である。

【図10】

図7のDVCR11-1の動作を説明するフローチャートである。

【図11】

図7のATM/1394中継器42-2の動作を説明するフローチャートである。

【図12】

図11のフローチャートの処理のタイミングを説明するタイミングチャートで

ある。

【図 13】

図 7 の ATM/1394 中継器 42-2 の中間バッファの構成例を説明する図である。

【図 14】

図 11 のステップ S57 のタイムスタンプ書き換え処理の詳細を説明するフローチャートである。

【図 15】

図 14 のステップ S87 の処理を説明するタイミングチャートである。

【図 16】

図 11 のステップ S61 のオーバーフロー防止処理の詳細を説明するフローチャートである。

【図 17】

図 11 のステップ S62 のアンダーフロー防止処理の詳細を説明するフローチャートである。

【図 18】

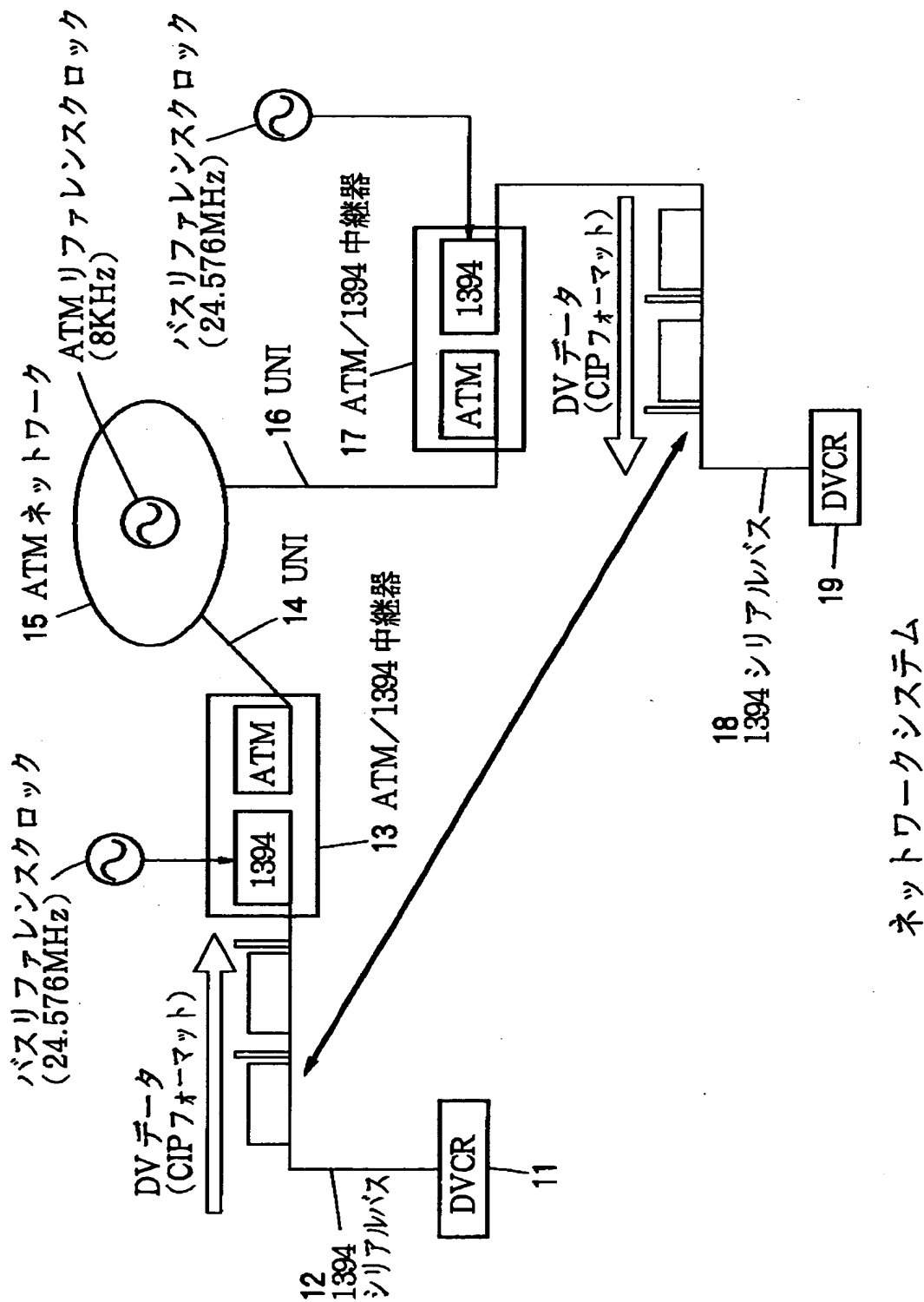
図 7 の DVCR19-2 の動作を説明するフローチャートである。

【符号の説明】

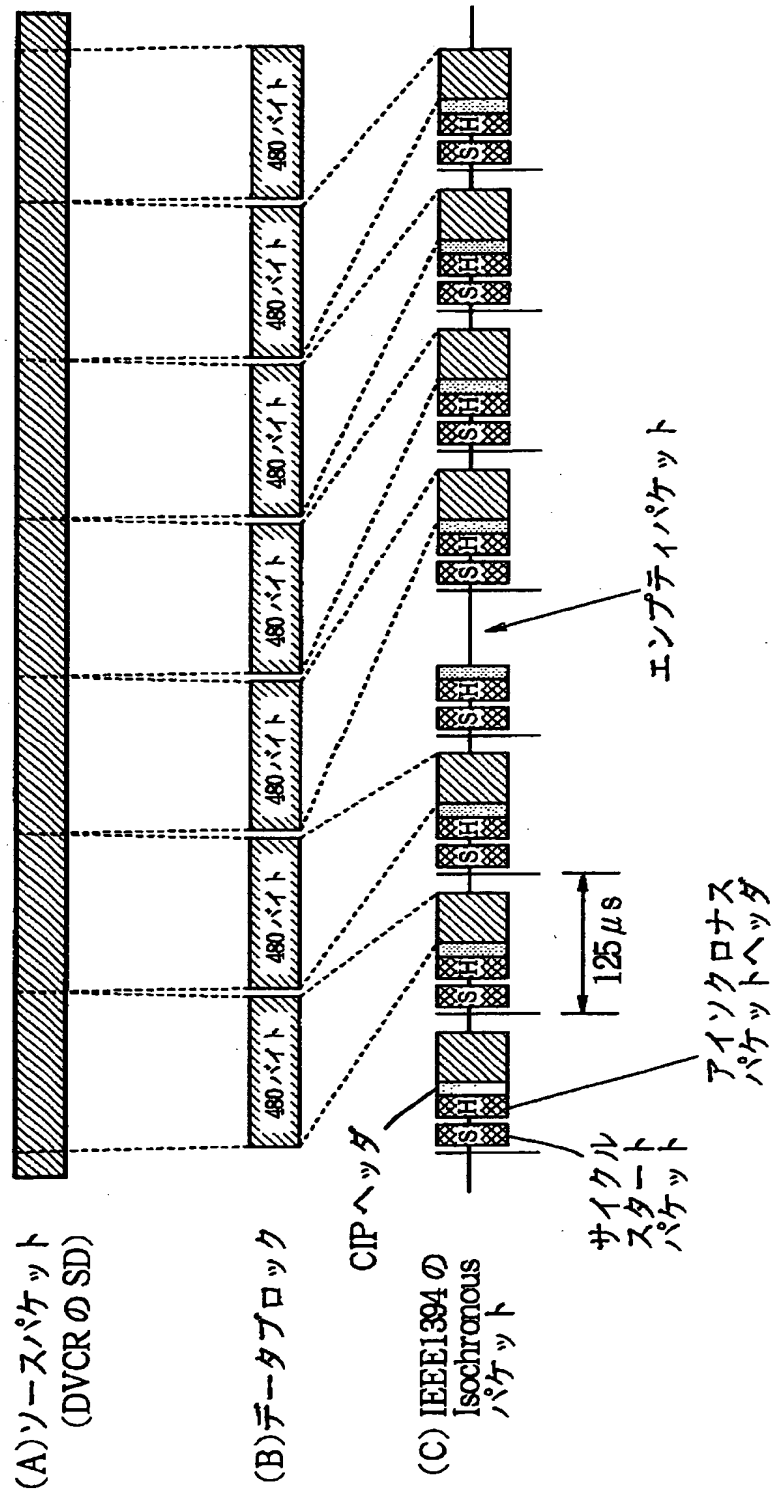
11-1, 11-2 DVCR, 12-1, 12-2 1394 シリアルバス, 15-1, 15-2 ATM ネットワーク, 18-1, 18-2 1394 シリアルバス, 19-1, 19-2 DVCR, 41-1, 41-2, 42-1, 42-2 ATM/1394 中継器, 59 クロック同期回路, 60 CPU, 62 メモリブロック, 63 スケジューリングタイマ, 71, 72 ブロック

【書類名】 図面

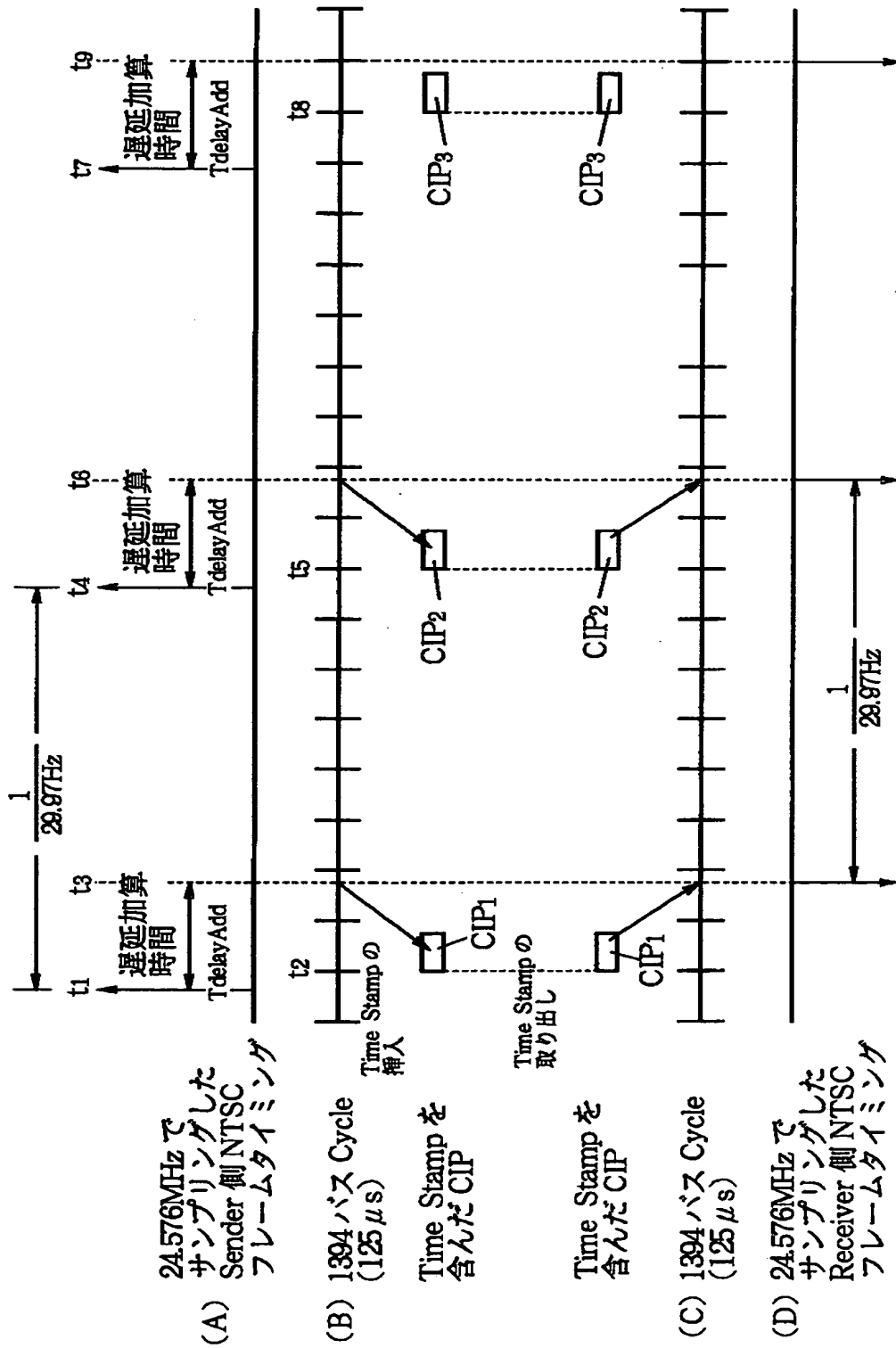
【図 1】



【図2】

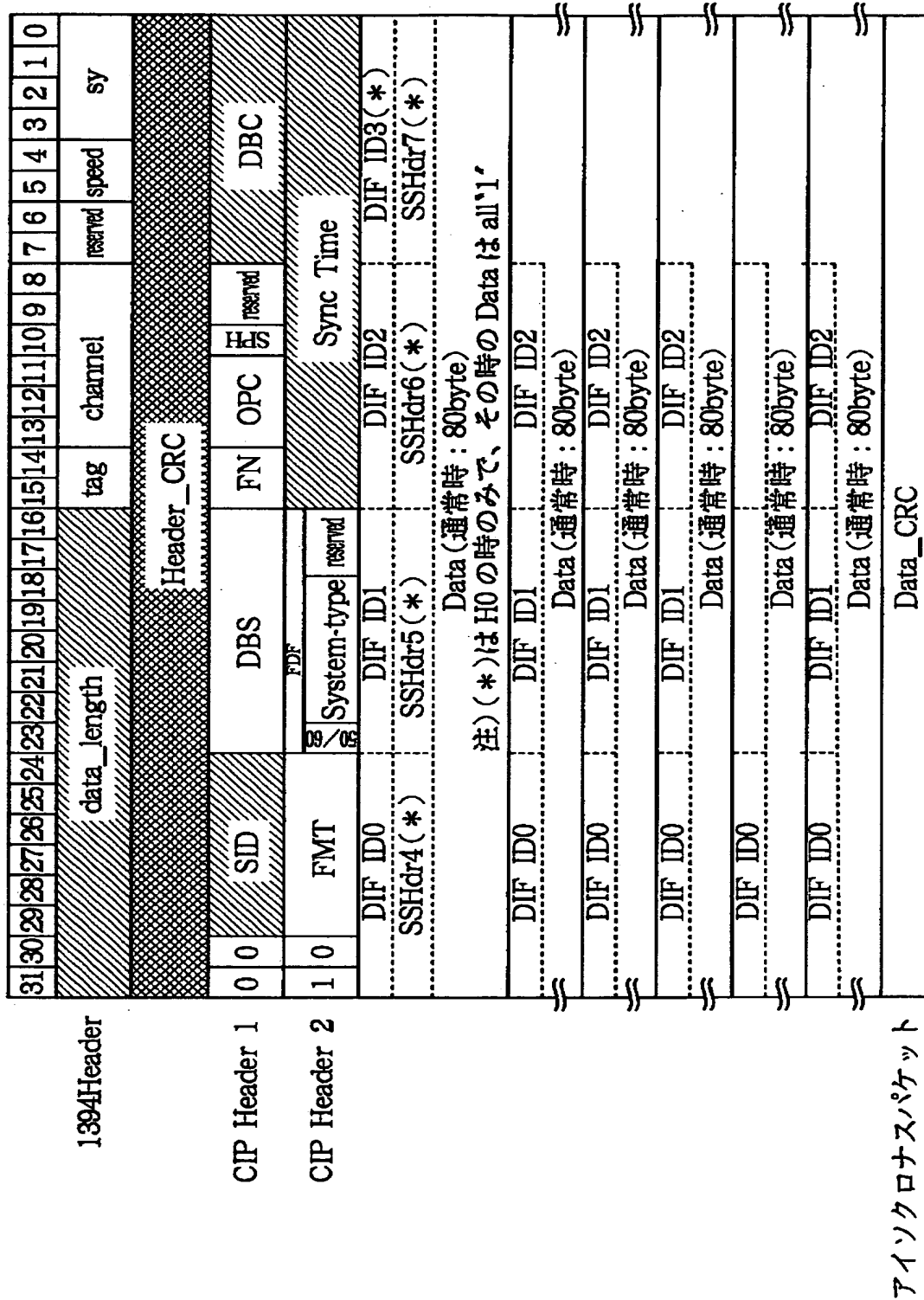


【図 3】

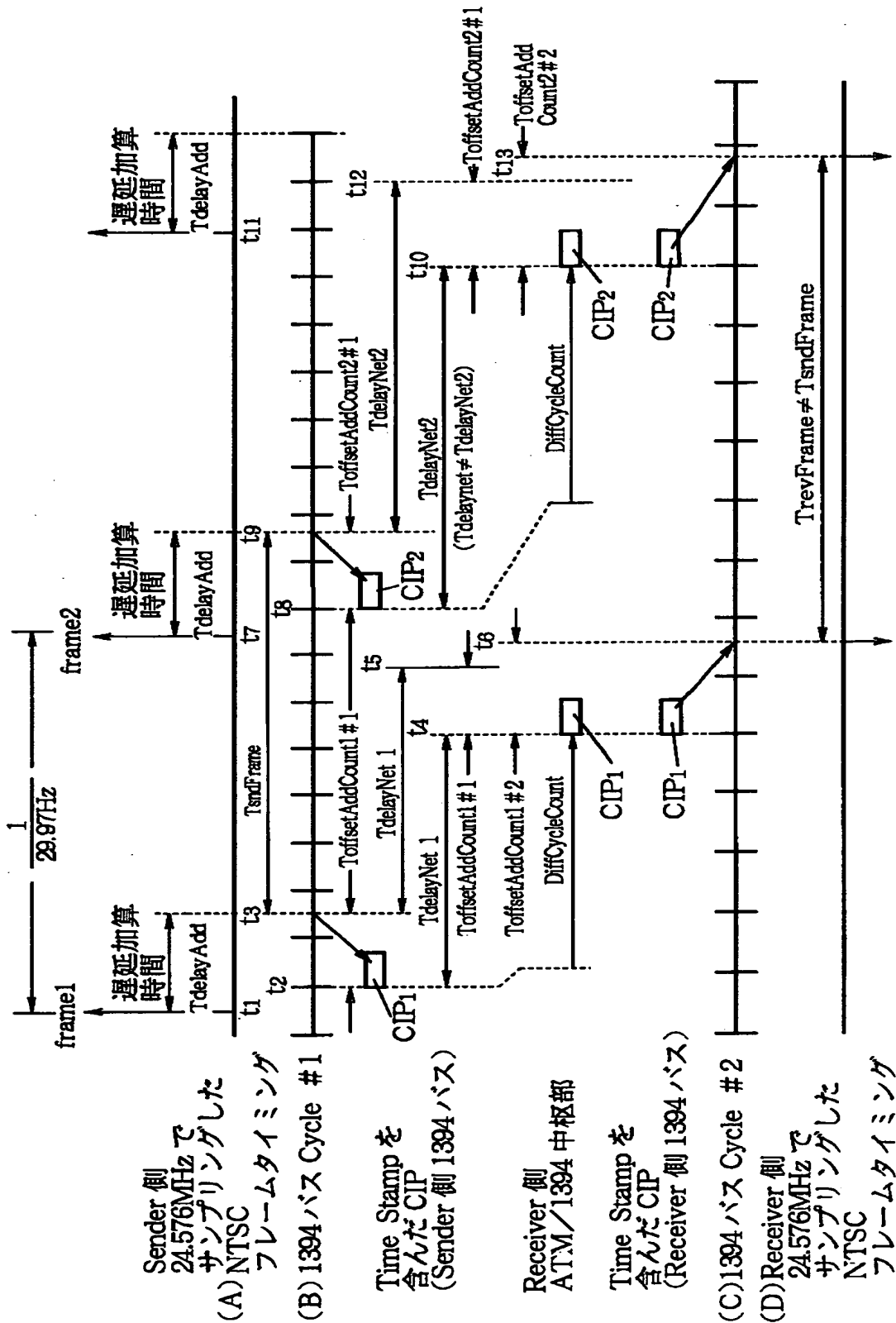




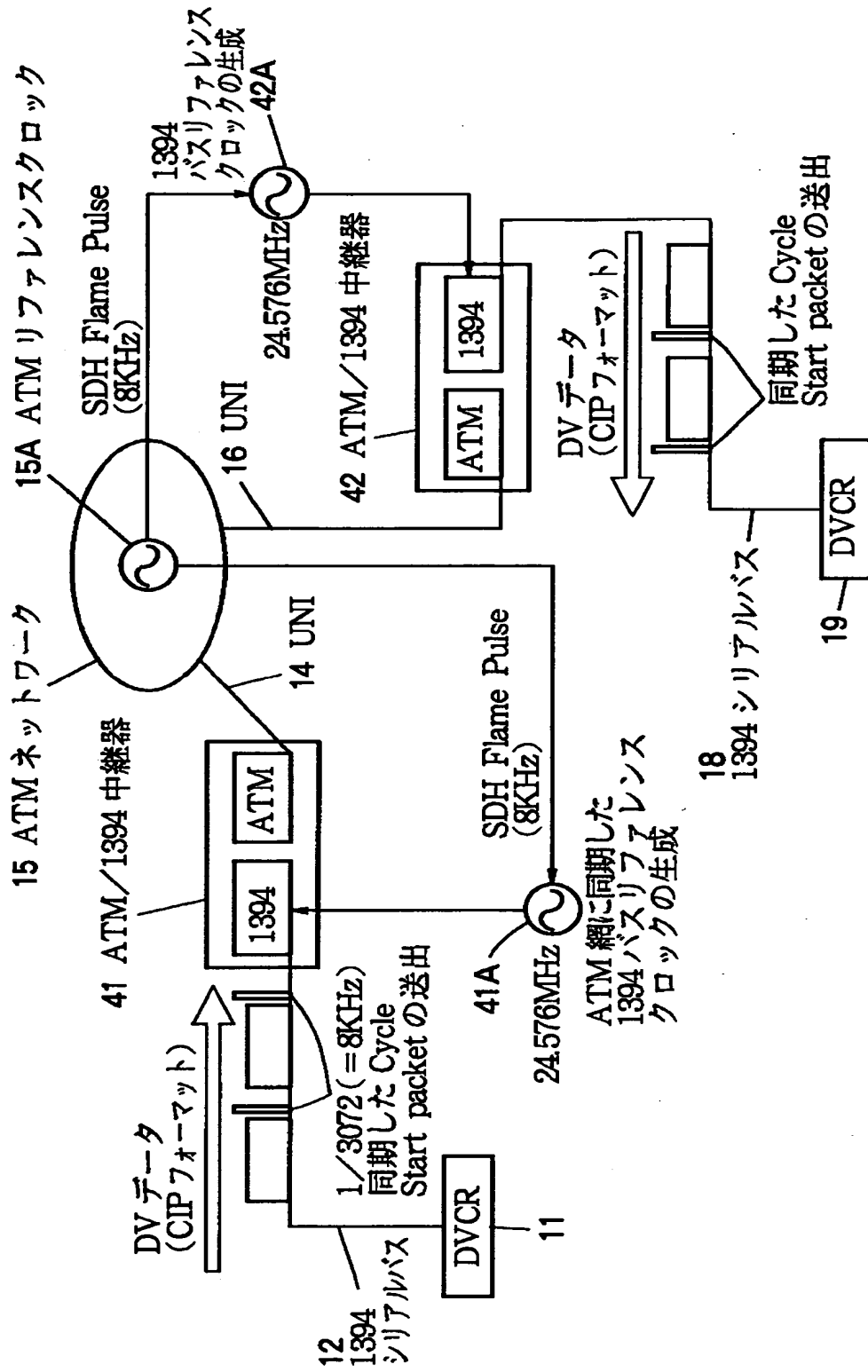
【図 4】



【図 5】

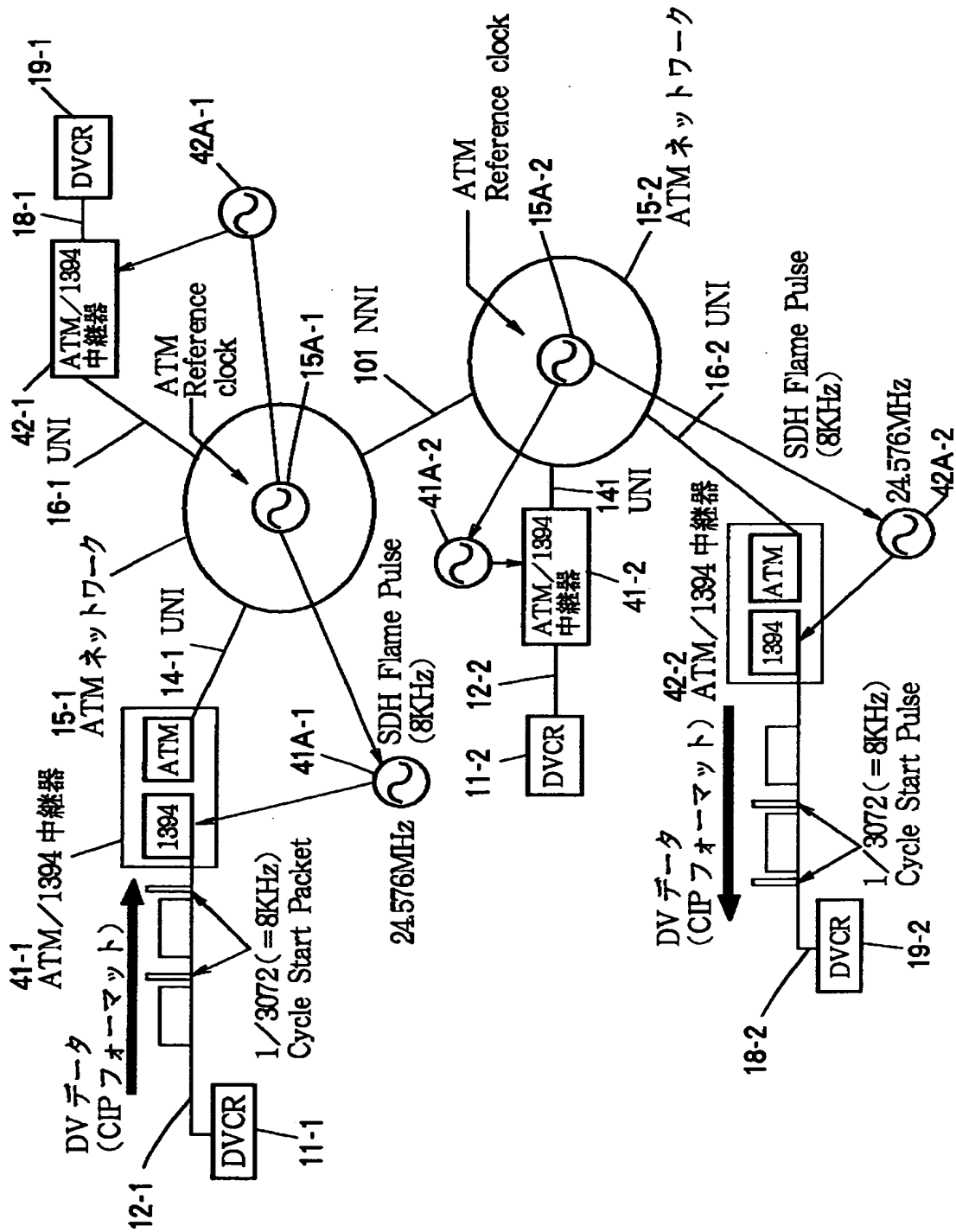


【図6】

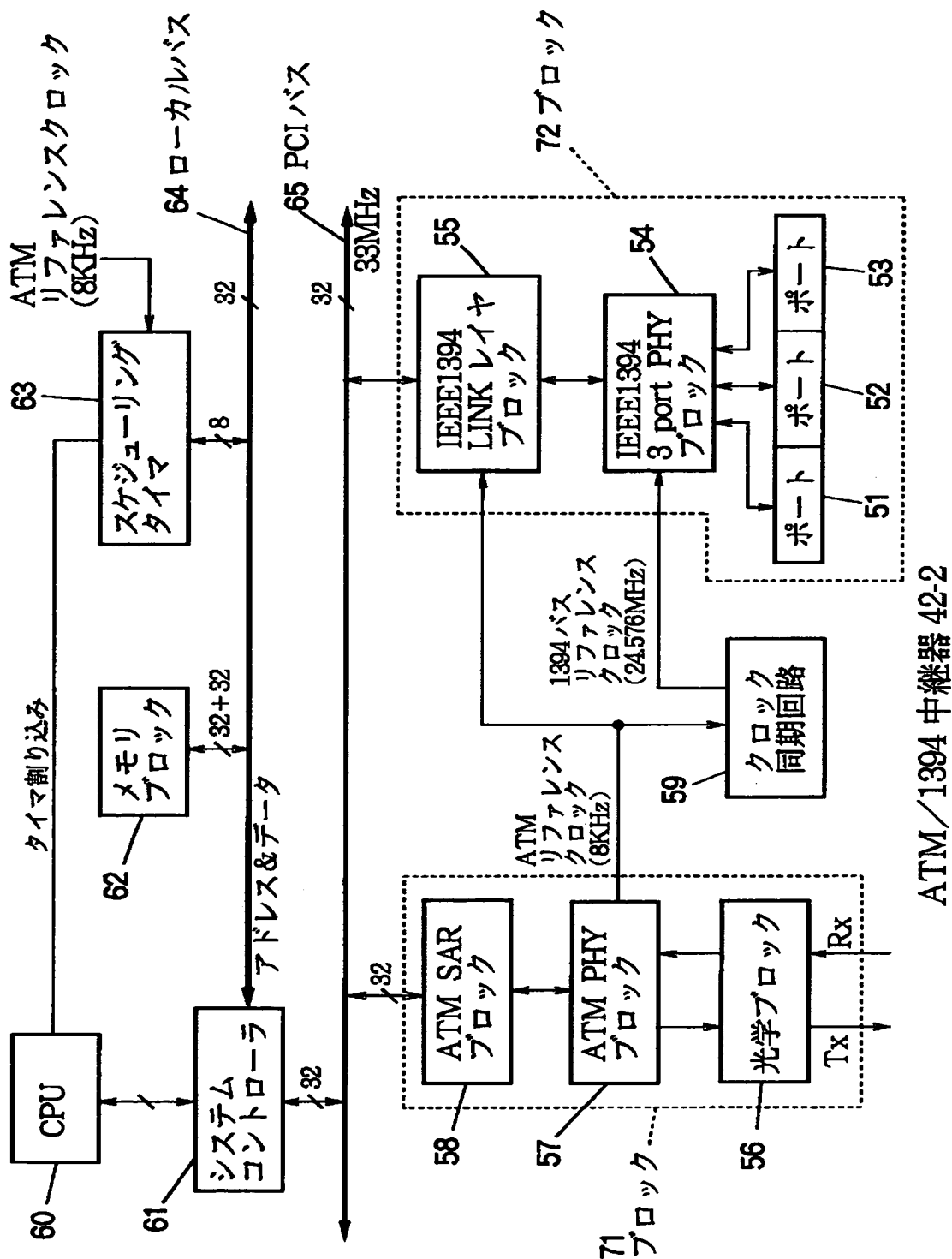


ネットワークシステム

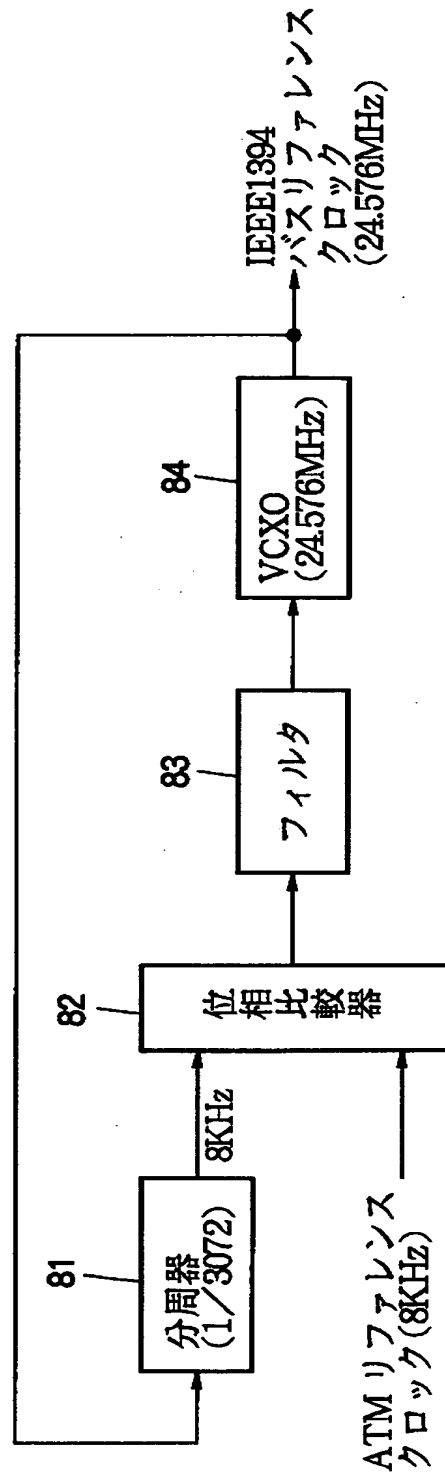
【図 7】



【图 8】

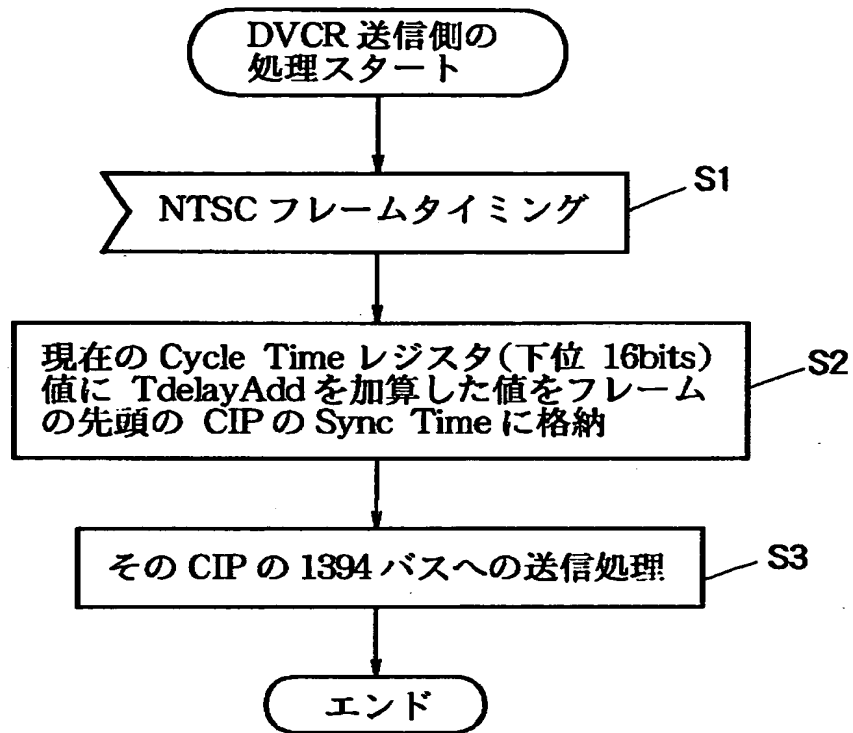


【図 9】

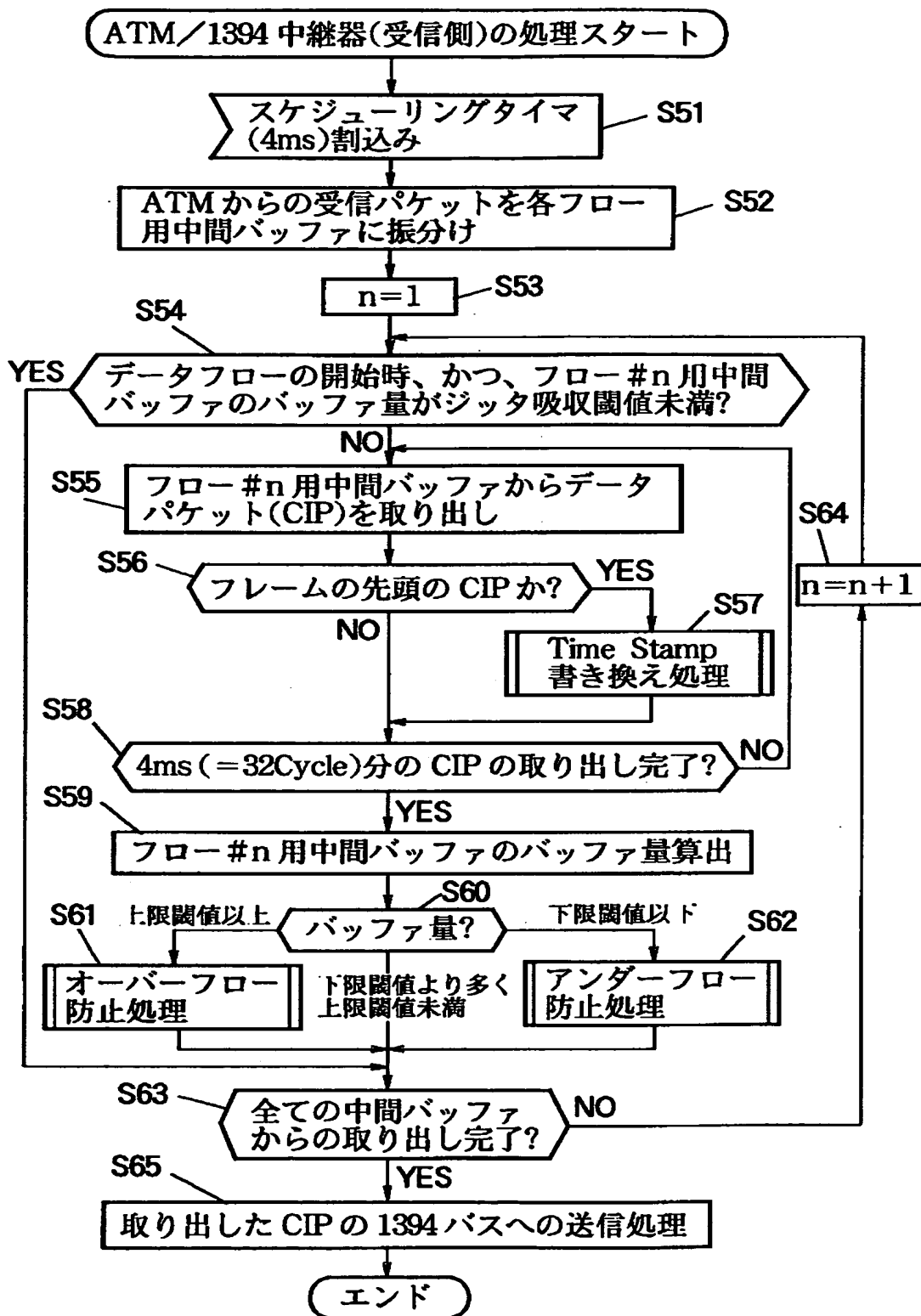


クロック同期回路 59

【図 1 0】

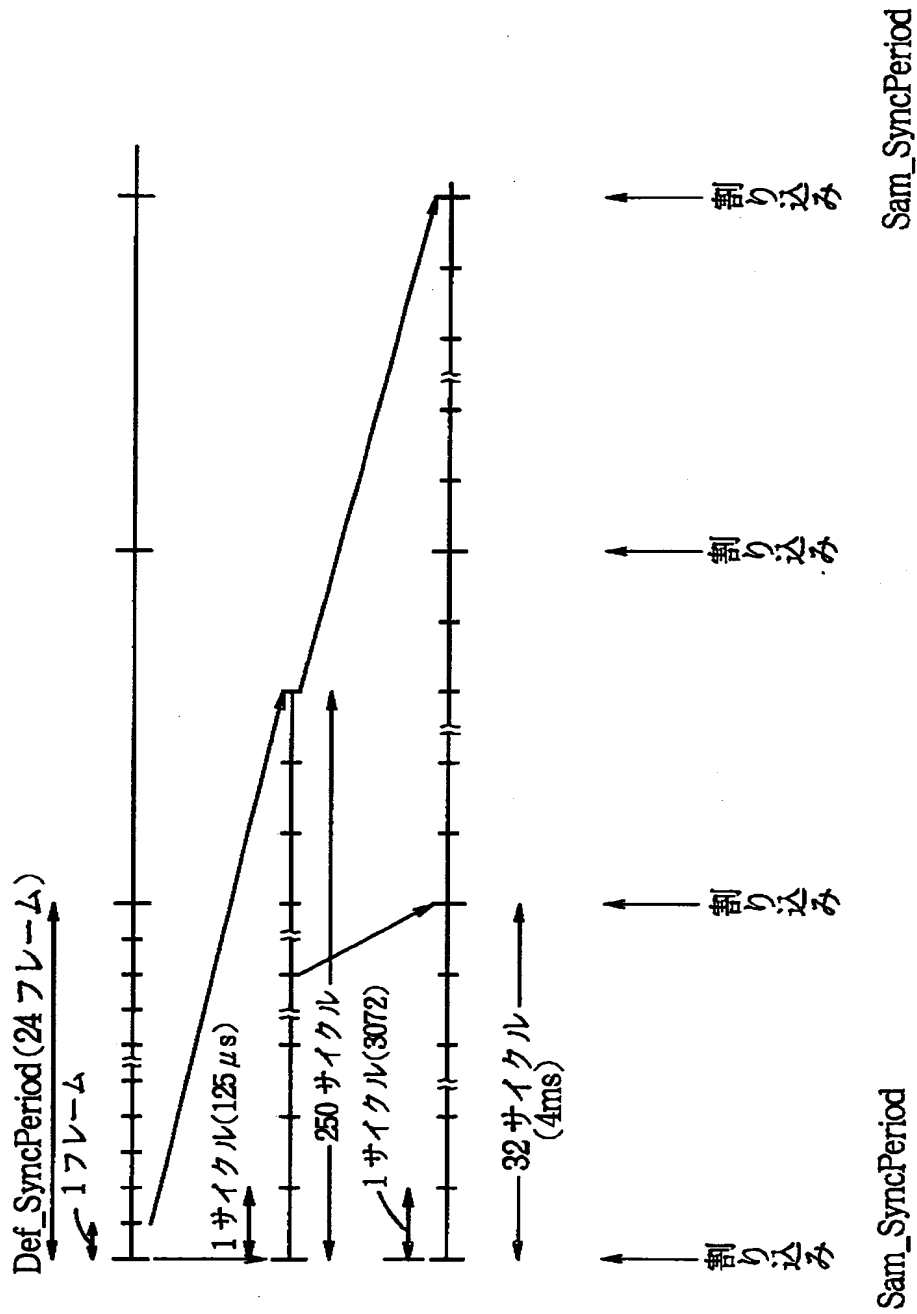


【図 11】

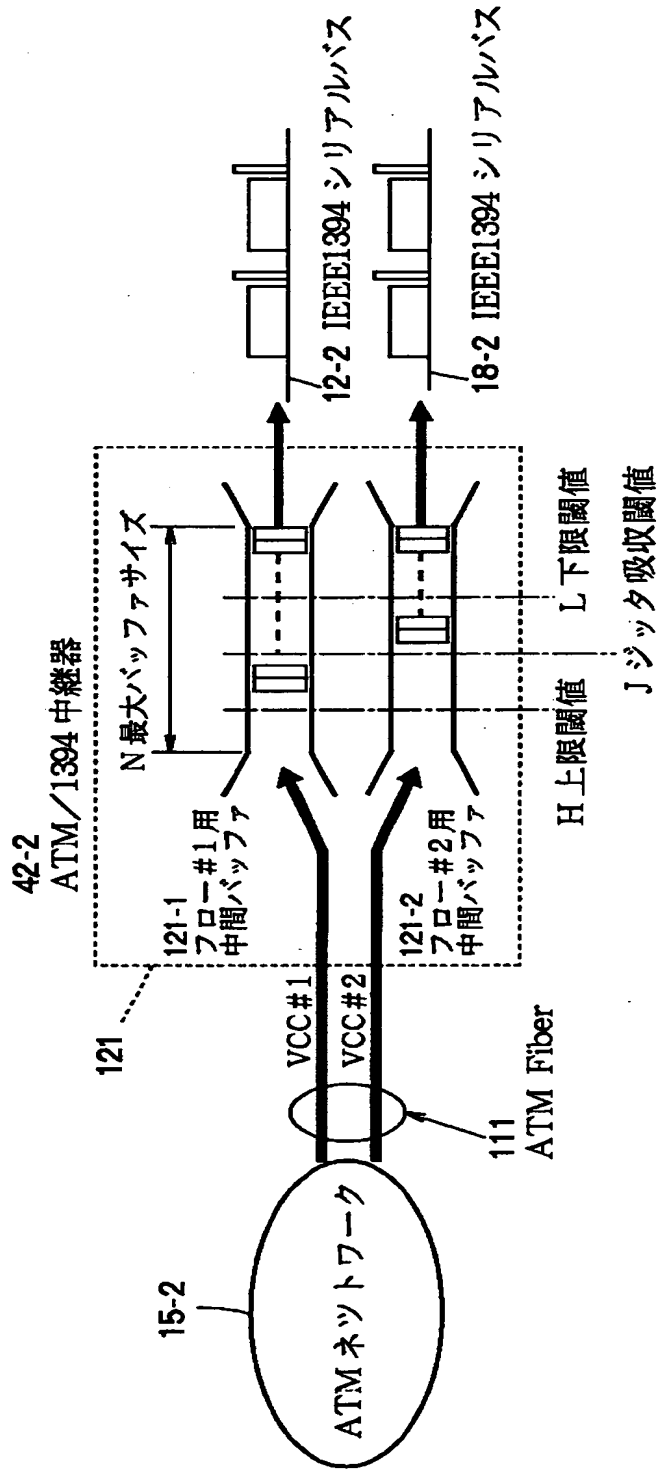




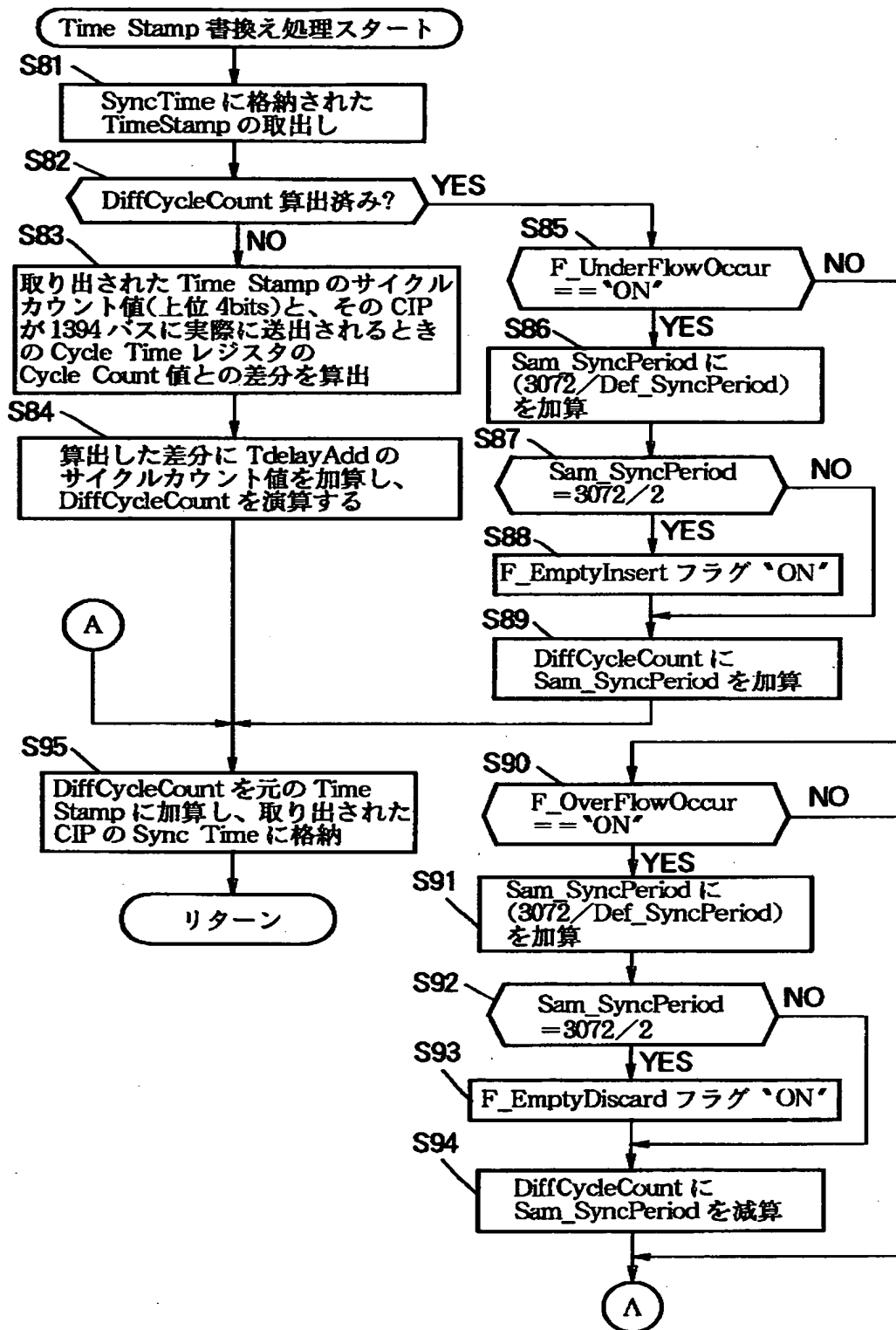
【図 12】



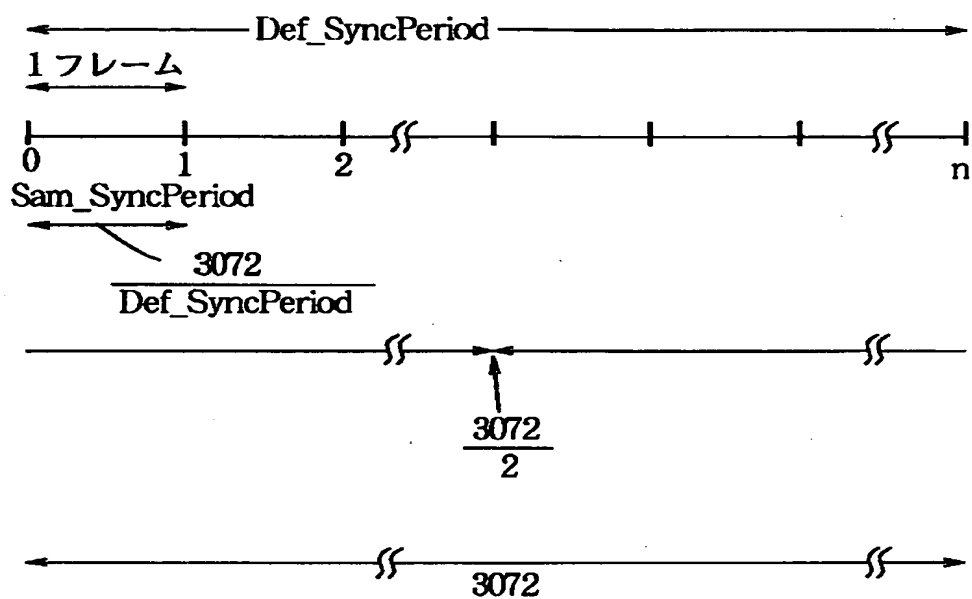
【図 13】



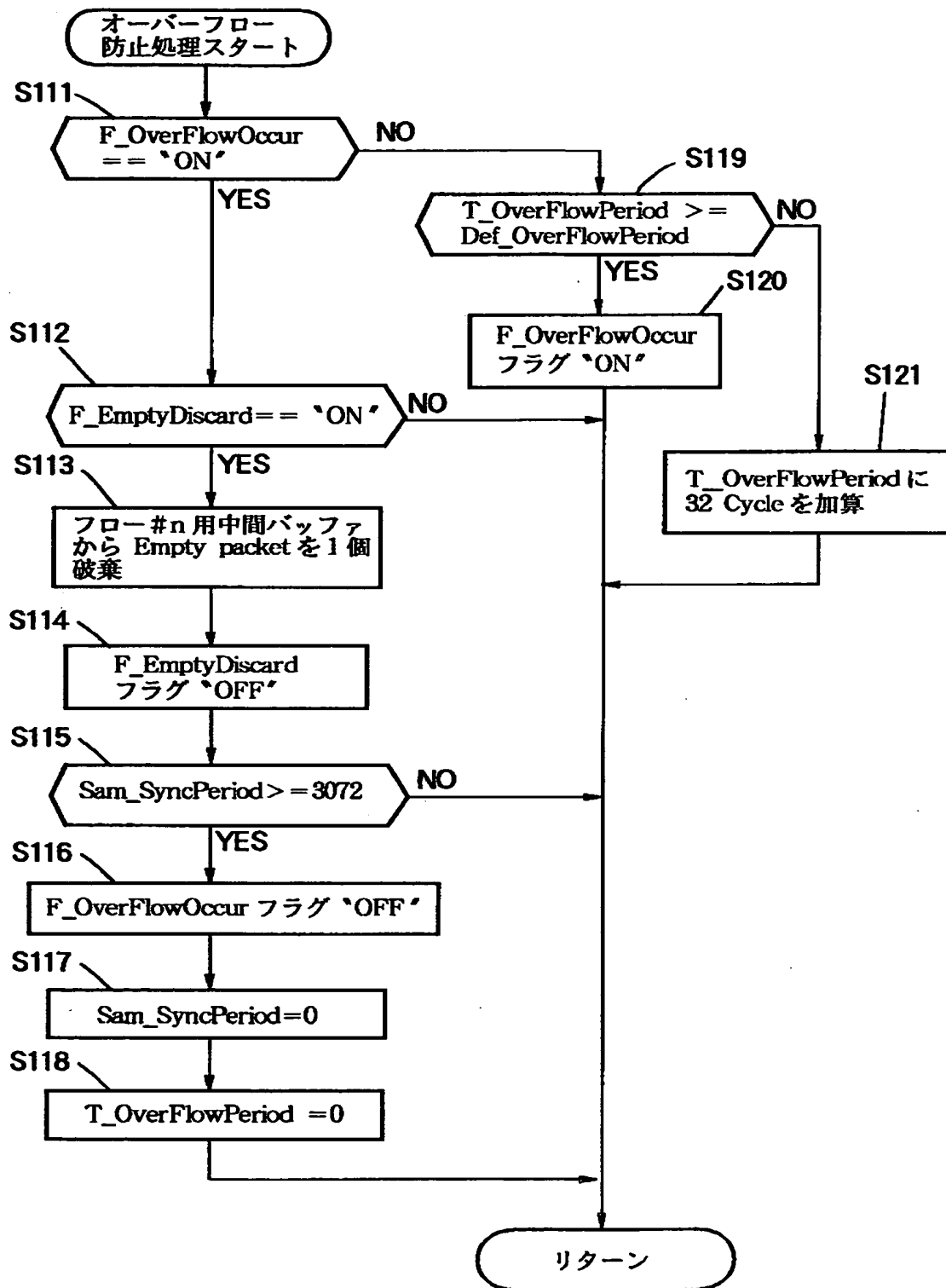
【図 14】



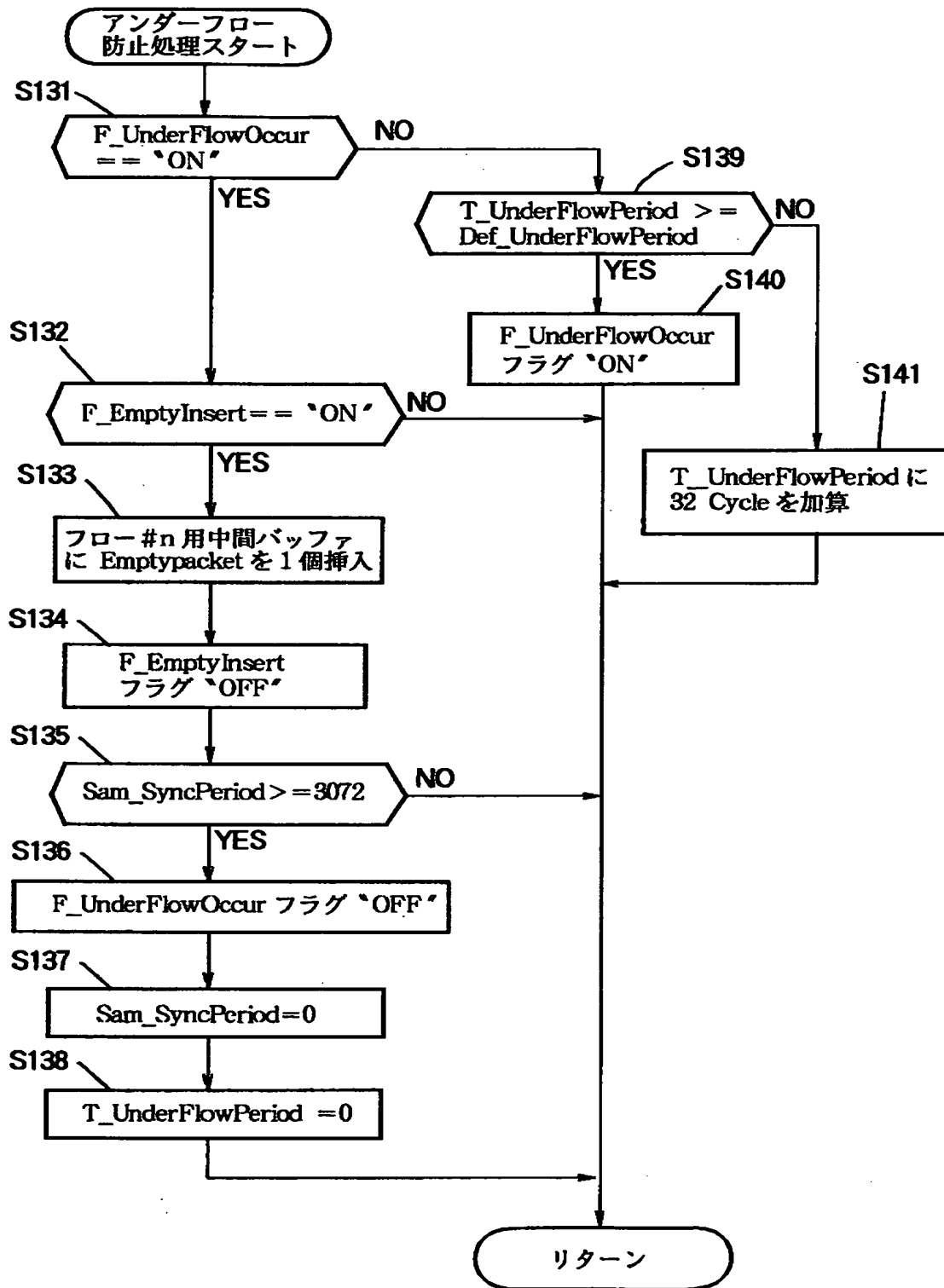
【図 1 5】



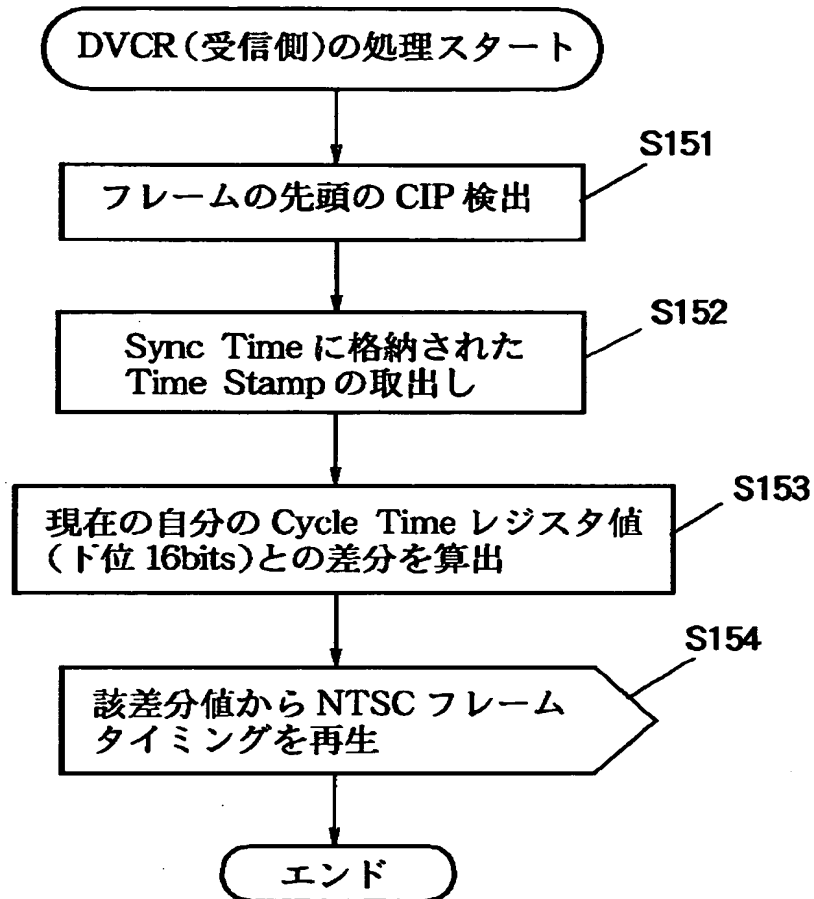
【図16】



【図 17】



【図 1 8】



特 2 0 0 0 - 0 3 8 8 7 1



【書類名】 要約書

【要約】

【課題】 非同期のリファレンスクロックで動作する複数のネットワークを介してデータを授受する場合のオーバーフローとアンダーフローを防止する。

【解決手段】 DVCR11-1より出力されたデータを、1394シリアルバス12-1、ATM/1394中継器41-1、UNI14-1、ATMネットワーク15-1、NNI101、ATMネットワーク15-2、UNI16-2、ATM/1394中継器42-2、1394シリアルバス18-2を介して、DVCR19-2に転送する。ATM/1394中継器42-2は、データを、フロー毎に、バッファに蓄積し、バッファの蓄積量に基づいて、エンプティパケットを削除するかまたは挿入することで、オーバーフローとアンダーフローを防止する。エンプティパケットの挿入または削除に伴って発せするタイムスタンプのズレは、徐々に補正される

【選択図】 図7

出 願 人 履 歴 情 報

識別番号 [000002185]

1. 変更年月日	1990年 8月30日
[変更理由]	新規登録
住 所	東京都品川区北品川6丁目7番35号
氏 名	ソニー株式会社